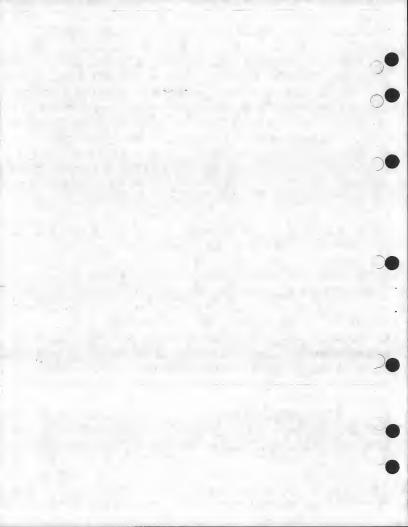
JUL 0 9 1976						ISSUE	NU	MBER	AND	DAT	E		_
CONTENTS 3-37	PAGE NUMBER	6-29-76											
PAGE INDEX	A1	1		-	-								
SUPPORTING INFORMATION										_			F
APPLICATION	B1	1	-										
	to												
	B4	1					_			_	-	-	-
													İ
COMMANDS	Cl	1				-		-	-	-	-	-	+
	C32	1	-	-	-	-							İ
		F			-	+							+
					F			-	-	-	-	-	+
													#
	-	-	-	-	-	+-	-	-	-	-	+		+
		1		1									I
THE CONTENT OF THIS MATER IT IS FURNISHED PURSUANT EXTENT OF DISCLOSURE, IT: PERMISSION OF ITS OWNER, I	TO WRITT	EN AG	REDIE	ITS O	RINS	TRUCTI	ONS	IT THE	WRIT	TEN			
PAGE INDEX NOTES				S	UPPO	RTIN	G 11	FOR	ATIO	M			
				CAT	EGOR	Y					NUM	BER	
1. WHEN CHANGES ARE MADE IN THIS DOCUMENT ONLY THOSE PAGES AFFECTED WILL BE REISSUED. 2. THIS PAGE INDEX WILL BE REISSUED AND BROUGHT UP TO DATE EACH TIME ANY PAGE OF THE DOCUMENT IS	MAIN		ANCE	REF	ERE	NCE				PK-	LC90	0	

SUPPURITING THE	OHI-INT TON
CATEGORY	NUMBER
MAINTENANCE REFERENCE MANUAL	PK-1C900
COMMON SYSTEMS	AT&TCo PROV
COMMAND MANUAL	'PK-1C901-01-A1 37 PAGES
	CATEGORY  MAINTENANCE REFERENCE MANUAL  COMMON SYSTEMS  3A CC



#### APPLICATION

### 1. INTRODUCTION

1.01 THE SECTIONS OF THIS SPECIFICATION CONSTITUTE THE COMMAND MANUAL FOR THE JACC.
ALL THE JACC COMMANDS AFF INCLUDED, TOGETHER WITH A DETAILED DISCRIPTION OF
EACH, AND GENEFAL INSTRUCTIONS FOR THEIR USE IN PROGRAMMING.

#### 2. DESCRIPTION

2.01 THIS SPECIFICITION DESCRIBES ALL THE COMMANDS IN THE BACK INSTRUCTION SET 2.01 THIS SPECIFICITION DESCRIEFS ALL THE COMMANDS IN THE BACK INSTRUCTION SET SO THAT THE PROGRAMMEN MY SELECT THE INSTRUCTIONS BEST SHITTED TO THE TARK SETHED PROGRAMMEN. THE FAMBLE CONTRINS A DETAILED DESCRIPTION OF EACH INSTRUCTION. AND THE CONTRINS THAT MUST BE SET UP PRIOR TO THEIR USE.

""" APPOYIMATE THE INTERVAL PEGGINED TO EXECUTE EACH COMMAND IS SUPPLIED FOR EACH.

### 4. GENERAL NOTES AND EXPLANATIONS

a.01 EACH COMMAND IS ILLUSTRATED AS IT APPEARS IN THE FULL PROGRAM STOPE WOPD.

CEPTAIN COMMANDS APE DOUBLE WOPD COMMANDS AND ARE SO DEPICTED.

4.02 IN THE LIMITATION OF THE PROGRAM STORE WORD, PM AND PL STAND FOR THE

A.03 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

4.03 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

5.03 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.04 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.05 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETITIVELY.

6.07 HALT HIGH AND PARLYT LOW BITS PERPETI COMSIDERED TO BE A BRANCH.

8.08 A MICROSTORE CYCLE IS DEFINED AS 150 NSEC.

4.05 THE OPERATION CODES THAT ARE SHOWN IN THE PROGRAM STORE WORD ILLUSTRATIONS ARE IN HEXADECMIAL.

4.06 CP CODF MNEMONICS BEGINNING WITH:

A	INDICATES	ADD
B	INDICATES	BRANCH
C	INDICATES	COMPARE OR COMPLEMEN
EX	INDICATES	EXCHANGE
I	INDICATES	INSERT
I	INDICATES	LOAD
24	INDICATES	MAINTENANCE
N	INDICATES	AND
C	INDICATES	INCLUSIVE OR
PL	INDICATES	ROTATE LEFT
PP	INDICATES	ROTATE RIGHT
S	INDICATES	SET OR SUBTRACT
ST	INDICATES	STORE
T	INDICATES	TEST
y	INDICATES	EXCLUSIVE OF
Z	INDICATES	ZERO

#### 4.07 SYMBOLS USED IN COMMAND DESCRIPTIONS:

CF	CONDITION FLIP-FLOP
#A	CONTENTS OF WORD IN MEMORY
**	NOT
8	OR
•	CVA

### APPLICATION

# 4.08 OPERANT SYMBOLS:

Px OF Py	ANY OF THE 16 GENERAL REGISTERS (P15-P0) WITHIN THE BACC. SEE PAGE C1.
Per	ANY OF THE 16 SPECIAL PEGISTERS WITHIN THE BACC. SEE PAGE C11 AND TABLE BELOW.
	A1 ** ADDRESS LIPPT PREISTER - ADDRESS MATCH FUNCTION AX ** ADDRESS MANY FEISTER - ADDRESS MATCH FUNCTION DB ** DEPART SUFFEE - PANDE LISPLAY DBY - DATA SHOW FEISTER - DATA MATCH FUNCTION DBY - DATA MARK REGISTER - DATA MATCH FUNCTION EX ** DERIVER FEISTER - DATA MATCH FUNCTION EX ** DATA SHOW FEISTER FEISTER IS ** INTERPUPE SET PREISTER (READ OMLY) MAINTERNACE CHANNEL TUPER REGISTER MISSER ** MAINTERNACE CHANNEL TRANSHITP METER MSSER ** MAINTERNACE STATES REGISTER MSS ** MAINTERNACE STATES REGISTER MSS ** MAINTERNACE STATES REGISTER SER ** STORT ADDRESS PREISTER SER ** STORT ADRESS PREISTER SER ** STORT ADDRESS PREISTER SER **
PA= PAO OP P12	GENERAL REGISTER ADDRESS PAIP OF P12 AND P13 USED TO CONTAIN A 20-BIT ADDRESS (BITS 3-0 OF P12 AND BITS 15-0 OF P13)
PA= PA1 CF P14	GENERAL REGISTER ADDRESS PAIR OF PIE AND RIS USED TO CONTAIN A 20-BIT ADDRESS (BITS 3-0 OF RIE AND BITS 15-0 OF RIS)
E I M N OFFSFT	BIT POSITION IMMEDIATE DATA MAST NUMBER (IMMEDIATE DATA OF INDEX) 6-BIT INDEX NUMBER LOW 8 BITE OF AN ADDRESS
X Y	MEMORY ADDRESS LOCATION

INSTE	OCT TON	TENGTH ,	RAGE NUM
AI Py	. I	D ALD 16 BITS OF IMMEDIATE DATA I TO DO	
All Px	"N	D ALD 16 BITS OF IMMEDIATE DATA, I, TO RK S ADD 4 BITS OF IMMEDIATE DATA, N, TO RK	17
AR Rx	.Fy		17
A1S Y		D ADD 1 TO THE CONTENTS OF MEMORY AT LOCATION V	17
b A		S PPANCH TO LOCATION Y	12
BCL Y		S PPANCH ON CONDITION TO LOCATION Y	12
BL. Y		D BRANCH LONG ON CONDITION TO LOCATION Y	13
BNC Y		D BRANCH LONG TO LOCATION Y S EFAMCH ON NOT CONDITION TO LOCATION Y	12
PNCL-		D BRANCH LONG ON NOT CONDITION TO LOCATION Y	13
PRAY		S EPANCH TO LOCATION RA INDEXED BY RX	13
BR N(	RA)		14 12
BRX F	x (FA)	S EPAPCH TO LOCATION RA INDEXED BY Rx	14
ESA Y			14
PSAI	x	S PRANCH AND SAVE ADDRESS INDIRECT	14
BTSA		S BRANCH TO SAVED ACDRESS	15
BTSAG		S GET REGISTERS 2 THROUGH 15 AND BRANCH TO SAVED ADDRESS	15
BTSAF		S GFT REGISTERS, LOAD RETURN CODE, AND BRANCH TO SAVED ADDRESS S LOAD RETURN CODE AND BRANCH TO SAVED ADDRESS	16
BX Px	- Y	S LOAD PETUPN CODE AND BRANCH TO SAVED ADDRESS D ERANCH ON INDEX NOT ZERO TO LOCATION Y	15
CI Px	.I	D COMPARE RX TO 16-BITS OF IMMEDIATE DATA, I	13
CIRM	Rx,I,N,M	D COMPARE 8 BITS OF PX ROTATED BY N WITH IMMEDIATE DATA AND MASK	21 21
COFL.	N(RA)	S COMPLEMENT WRITE THE OFF-LINE STORF AT LOCATION RA INDEXED BY N	30
COFLY	Rx (RA)	S COMPLEMENT WRITE THE OFF-LINE STORE AT LOCATION DA INDEVED DE DE	20
COM B	x[ , Py ]	S COMPLEMENT REFRY   AND STORE IN PE	₹ 19
CONL	N(RA)	S COMPLEMENT WRITE THE ON-LINE STOPE AT LOCATION DA INDEXED BY M	29
CCNIJ	Rx (RA)	S COMPLEMENT WRITE THE ON-LINE STORE AT LOCATION RA INDEXED BY RX	29
CR Ry	x,Pv,M	S COMPARE PY TO RX D COMPARE PY TO RX INDER 16 BIT IMMFDIATE MASK	20
FXP R	v. Du	S EXCHANGE THE CONTENTS OF RX WITH THE CONTENTS OF RY	21
FLZ R	x. Rv	S FIND LOW ZFRO IN PX AND RECORD ITS POSITION IN RY	9
GA		S GET REGISTERS 2 THROUGH 15 FROM WORDS 2 THROUGH 15 OF HOLD-GET AREA	22
GN Rx	. N	S GFT RX FPOM WOPD N OF HOLD-GET AREA	8
HA		S HOLD REGISTERS 2 THROUGH 15 IN WORDS 2 THROUGH 15 OF HOLD-GET APPA	7
HALT		S HALT THE CENTRAL CONTROL	32
HN Rx	"N	S HOLD RX IN WORD N OF HOLD-GET APPA	8
ICF P	x*ss	S INSERT CF IN BIT N OF RX	25
I RM R	x.Ry.M	D INSERT RY INTO PX UNDER IMMEDIATE MASK, M	9
L Rx,	, N (RA)	S LOAD RX WITH THE CONTENTS OF NEMORY AT LOCATION RA INDEXED BY N S LOAD RX FROM MEMORY AT LOCATION RA INDEXED BY N AND URDATE RA	2 2
LAT. P	X,Y,RA	D LOAD PR WITH THE CONTENTS OF MEMORY AT LOCATION Y AND SET RA TO Y	2
	x,Ry(RA)	S LOAD RE FROM MEMORY AT LOCATION RA INDEXED BY RY AND UPDATE RA	2
LI Px	.1	D LOAD RX WITH 16 BITS OF IMMEDIATE DATA, I	3
LL Rx	· Y	D LOAD RX WITH THE CONTENTS OF MEMORY AT LOCATION Y	1
LN Px	· N	S IOAD Rx WITH & BITS OF IMMEMENTATE DATA. N	3
LR RE	"Ry	S LOAD RE WITH THE CONTENTS OF RW	9
LRM F	x, Py, M	D LOAD RE WITH THE CONTENTS OF RV UNDER IMMEDIATE MASK. M	9
LRS R	x,Re		10
LSR R	S,RX	S LOAD RS WITH THE CONTENTS OF RX	10
MI	,Ry (RA)	S LOAD RX WITH THE CONTENTS OF MEMORY AT LOCATION RA INDEXED BY RY D MICPO INTERPRET	2
MIS		D CIMIT CACIF MICEO CHERDON	32
MSTF	N (RA)	D SINGLE CYCLE MICRO INTERPRET  D MAINTERANCE STORE FUNCTION USING REGISTER 0 AT LOCATION RA INDEXED BY  D MAINTERANCE STORE SUMMITTAN HERMS DEGISTER 0 AT LOCATION RA INDEXED BY	N 31
MSTFX	Rx (RA)	D MAINTENANCE STORE FUNCTION USING REGISTER 0 AT LOCATION RA INDEXED BY	Px 31
NI Px	.I	D AND 16 BITS OF IMMEDIATE DATA, I, TO RE AND STORE IN RE	19
POP		S NO OPERATION	32
NR Px	, Ry	S AND RY TO RE AND STORE IN RE	19
OI Rx	.I	D INCLUSIVE OF 16 BITS OF IMMEDIATE DATA, I, TO RX AND STORE IN RX	19
OR PK	, RY	S INCLUSIVE OR BY TO BY AND STORE IN BY	20
PACK	kle	S PACKS GENERAL REGISTERS 2 AND 3 INTO 20 BIT SPECIAL PEGISTER RS	11
RL Px	Das	S PPOGRAM INTERRUPT END	16
RLN P	w 10	S ROTATE RX LEFT AN AMOUNT DETERMINED BY THE LOW & BITS OF RY S ROTATE RX LEFT BY N BIT ROSITIONS	21
RR Rx	- Pu	S ROTATE RX RIGHT AN AMOUNT DETERMINED BY THE LOW & BITS OF RY	22
PRN P	x.N	S POTATE RX RIGHT BY N BIT POSITIONS	22
SBN R	x.B	S SET BIT 9 IN RX	23
SPR P	x, Py	S SFT BIT IN Fx DETERMINED BY THE LOW & BITS OF RY	23
SBS N	(RA),B	S SET RIT R IN MEMORY MORD AT LOCATION DETERMINED BY ADDING N TO DA	23
SCF	_	S SET THE CONDITION-FLOR	23
SI Px	,1	D SUPTRACT 16 BITS OF IMMEDIATE DATA, I, FROM RX	18
SMIC			27
SN Px	- 10	S SEND MAINTENANCE I/O MESSAGE OVER CHANNEL AND SUBCHANNEL DEFINED IN R9 S SUBTRACT & BITS OF IMMEDIATE DATA, N, FFOM Rx	
SOP	***	S SET OF CODE FIL BIT	18 23
SP Px	, Ry	S SUBTRACT RY FROM RX AND STORE THE RESULT IN PX	18
ST Rx	,N(RA)	S STORE PK IN NEMORY AT LOCATION PA INDEXED BY N	4
STA R	x,N(RA)	S STORE PK IN MEMORY AT LOCATION RA INDEXED BY N AND URDATE RA	
STAF	N(RA)	D STORE ACCESS FUNCTION USING REGISTER 0 AT LOCATION PA INDEXED BY N	5
STAFX	Rx (RA)	D STORF ACCESS PUNCTION USING REGISTER O AT LOCATION DA INDEVED BY DE	6
STAT.	Rx,Y,RA	D STOPE PX IN MEMORY AT LOCATION Y AND SET RA TO Y S STORE PX IN MEMORY AT LOCATION RA INDEXED BY RY AND UPDATE PA	4
STAX	Rx , Py (RA)	S STORE PK IN MEMORY AT LOCATION RA INDEXED BY RY AND UPDATE PA	5
STL F	~ * *	D STORE RI IN MEMOPY AT LOCATION Y	4

# 3A CC --- INSTRUCTION SET INDEX

INSTRUCTION	LFNGTB	PAGE NUMBER
STM Px.N(PA) .M	D INSEPT Px UNDER MASK INTO MEMOPY AT LOCATION RA INDEXED BY N	
STVH PX,N (PA)	S INSERT RX UNDER VARIABLE MASK INTO MEMORY AT LOCATION RA INDEXED BY N	6
STX Px, Py (RA)	S STOPE PX IN MEMORY AT LOCATION RA INDEXED BY RY	6
TPN Rw. P	S TEST BYT B TW D-	26
TBP Px. Pv	S TEST BIT IN BY DETERMINED BY JOH & BITC OF BU	26
TRS KIRAL - B	S TROP BIT B IN MEMORY MADE AT LOCATION COMPANY OF THE PROPERTY  26	
TCC1	S TEST BIT IN REDETERMINED BY LOW & BITS OF PY S TEST BIT B IN MEMORY WORD AT LOCATION DETERMINED BY ADDING N TO RA S TEST CENTRAL CONTROL 1	26
TCH	S TEST THE MAIN I/O CHANNEL DEFINED IN R9 FOR THE IDLE STATE	26
TIO	S TEST FOR I/O MESSAGE IN CHANNEL DEFINED IN R9	28
TMIO	S TEST FOR MAINTENANCE I/O MESSAGE IN CHANNEL DEPINED IN PA	27
TPPH Px	S TEST GENERAL REGISTEP PARITY HIGH	28
TPPL Px	S TEST GENERAL REGISTER PARTY LOW	25 25 25 25 25
TSPPF PR	S TEST SPECIAL PEGISTEP PARITY HIGH	25
TSPPI Fs	S TEST SPECIAL REGISTER PARITY LOW	25
TZ Px	S TEST RE FOR ALL ZEROS	25
UNPF Rs	S UNPACKS 20 BIT SPECIAL PEGISTER PS TO GENERAL PEGISTEPS 2 AND 3	22
XI Px,I	D EXCLUSIVE OF 16 BITS OF IMMEDIATE DATA, I, TO RE AND STOPE IN RE	11
XP Px,Py	S EXCLUSIVE OR PY TO RX AND STORE IN RX	20
7BN Px,B	S 72PO BIT B IN Rx	20
ZBR Px, Py	S ZERO BIT IN PX DETERMINED BY THE LOW & BITS OF RY	24
ZBS N(PA) ,B	S ZERO BIT IN PR DETERMINED BY THE LOW & BITS OF RY	24
ZCF		24
710		24
TOP	S ICLE THE MAIN I/O CHANNEL DEFINED IN P9 S ZEPO OP CODE FIL RIT	28
7.OF	S ZEPO OP CODE FIL BIT	24

DEFINITION OF ADDRESSABLE 16-BIT GENERAL PURPOSE REGISTERS

n	1	REGISTER 0
1	1	REGISTER 1
2	1	REGISTER 2
3	1	REGISTER 3
4	ı	REGISTER 4
5	1	REGISTER 5
6	1	REGISTEP 6
7	1	REGISTER 7
8	- I	REGISTEP 8
9	1	REGISTER 9 ALSO USED FOR IMPUT/OUTPUT
10	ı	REGISTER 10 ALSO USED FOR INPUT/OFFPUT
11	1	REGISTEP 11 ALSO USED FOR INPUT/OUTPUT
12	-1	REGISTER 12 ALSO USED FOR MEMORY ADDRESSING
13	ı	REGISTER 13 ALSO USED FOR MEMORY ADDRESSING
14	1	REGISTEP 14 ALSO USED FOR MEMORY ADDRESSING
15	1	REGISTEP 15 ALSO USED FOR MEMORY ADDRESSING

17 | 16 | 15 | 10 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 0 | 3 | 2 | 1 | 1

LL RX.Y LOAD RX WITH THE CONTENTS OF MEMORY AT LOCATION Y

í	PĦ	ı	PL	1	BA	1					0	PC	ODE	3	1						1				Rx			_	1	В	ITS	(19	-16)	OF	¥		1
ij.	PH	1	PL	ı					_			,							BIT	3 (1	5-0	) 0	PY				,										1
ĭ	17	- 1	16	1	15	1	14	1	13	1	12	1	11	1	1	0	1	9	1	8	1	7	1	6	1	5	1	4	ı	3	1	2	1	1	1	0	

<sup>1.</sup> LOAD Rx WITH THE CONTENTS OF MEMORY AT LOCATION Y.

APPROXIMATE EXECUTION TIME 3.75 MICROSECONDS

<sup>2.</sup> MEMORY AT LOCATION Y IS UNCHANGED.

LAL PX.Y.PA

LCAP PK WITH THE CONTENTS OF MEMORY AT LOCATION Y AND SET RA TO Y

PH	1	PI,	1	BA	1 (	P CODES	32/33	1	DX	1	BITS (19-16)	OF Y
PH		Pf.	1					BITS (15-0) OF Y				

1 17 | 16 | 15 | 18 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 |

1. LOAD RK WITH THE CONTENTS OF MEMOPY AT LOCATION Y AND SET RA TO Y.

2. Px SHOULD NOT BE EQUAL TO EITHER MEMBER OF THE RA REGISTER PAIR.

3. MEMOFY AT LOCATION Y IS UNCHANGED.

NOTE: BIT 8 IS DETERMINED BY THE VALUE OF DA. IT IS 0 FOR RA=12 AND 1 FOR RA=14. BITS (15-4) OF GENERAL REGISTER 12/14 ARP NOT CHANGED BY THIS INSTRUCTION. SEE NOTE CONCEPNING RA ON PAGE 82.

APPROXIMATE EXECUTION TIME 3.90 MICROSECONDS

L. Ry. N(PA)

LOAD RK WITH THE CONTENTS OF MEMORY AT LOCATION RA INDEXED BY N

1	P	н	ī	PL	1	BA	1					01	co	DE	3 4	10/	41					1				Rx				1				19				j
	1	17	ī	11	5 1	1	5 1	1	8	1	3		12		11	1	10	1	9	ī	8	1	7	1	6	-	5	1	4	1	3	1	2	-	1	1	0	1

1. LOAD Px WITH THE CONTENTS OF MEMORY AT LOCATION DETERMINED BY ADDING N TO RA.

NOTE: BIT 8 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR PA=12 AND 1 FOR RA=14. SEE NOTE CONCEPNING PA ON PAGE B2.

APPROXIMATE EXECUTION TIME 2.55 MICROSECONDS

LA RK,N(RA)

LOAD RX FROM MEMORY AT LOCATION RA INDEXED BY N AND UPDATE RA

í	PH	-	1	PL	1	E	IA.	1	_		_	-	0	P C	ODE	es t	12/	43	_		 	_	1	_			Rx	_	_	_	1		_	_	14				٦
1	1	7	1	1	5 [	_	15	ī	14	1	1	3 [	_	12	1	11	1	10	1	9	1	8	1	7	-1	6	1	5	1	4	-1	3	1	2	1	1	-1	0	-

1. LOAD RX WITH THE CONTENTS OF MEMORY AT LOCATION DETERMINED BY ADDING W TO RA AND UPDATE RA.

2. Rx SHOULD NOT BE EQUAL TO EITHER MEMBER OF THE RA REGISTER PAIR.

NOTE: BIT 8 IS DETFRMINED BY THE VALUE OF RA. IT IS 0 FOR RA=12 AND 1 FOR RA=14. BITS (15-4) OF GENERAL PEGISTER 12/14 ARE NOT CHANGED BY THIS INSTRUCTION. SEE NOTE CONCERNING PA ON PAGE B2.

APPROXIMATE EXECUTION TIME 2.55 MICROSECONDS

LX Rx. Rv (RAI

LGAD Rx WITH THE CONTENTS OF MEMORY AT LOCATION RA INDEXED BY Ry

1	P	Н	1	PL	-	BA	- 1					CP	COL	ES.	44/	45			T		1				Px				1				Ry				j
	ı	17	1	1	5 1	1	5 1	14	1	13	1	12	1	11	1	10	1	9	-	8	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	1

1. LOAD RX WITH THE CONTENTS OF MEMORY AT LOCATION DETERMINED BY ADDING RY TO RA.

NOTE: BIT 8 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR RA=12 AND 1 FOR RA=14. SEE NOTE CONCEPNING RA ON PAGE B2.

APPROXIMATE EXECUTION TIME 2.55 MICROSECONDS

LCAD Px FROM MEMORY AT LOCATION RA INDEXED BY RY AND UPDATE RA LAX Px, Py (PA)

PH   PL   BA	OP CODES 46/47	l Rx	1	Pv	
1 17 1 16 1 15 1	14   13   12   11   10   9	1 0 1 7 1 6 1 6 1			

- 1. LOAD Px WITH THE CONTENTS OF MEMOFY AT LOCATION DETERMINED BY ADDING RY TO PA AND UPDATE RA.
- 2. Px SHOULD NOT BE EQUAL TO EITHEP MEMBER OF THE PA REGISTER PAIR.

NOTE: BIT 8 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR PA=12 AND 1 FOR PA=18. BITS(15-8) OF GENERAL PEGISTER 12/18 AFF NOT CHANGED BY THIS INSTRUCTION. SEE NOTE CONCEPTING RA OR PAGE 32.

APPROXIMATE EXECUTION TIME 2.55 MICROSECONDS

#### LI Px,I LOAD Px WITH 16 BITS OF IMMEDIATE DATA, I

PH	- 1	PL	1	BA	1				OP (	CODE	01						1				Rx				1			 1	_			7
PR	1	PL	1													_	I			_			_						_			1
1 17	7 1	16	1	15	1 14	1	13 (	1:	2 1	11	-	10	-	0.	-	-		7	-	-	_	-		-	_	_	_	 _		_	_	-4

1. LOAD Px WITH I.

APPROFIMATE EXECUTION TIME 2.40 MICROSECONDS

LN Rx.N LOAD Rx WITH & BITS OF IMMEMEDIATE DATA, N

í	PH	1	P	L	1	BA	1.			OP	cc	DE	06			-			1	_	_		Rx		_		1	_			N			_	٦
ī	1	7 1	1	16	8	15	1 1	4 1	13	12	ī	11	1	10	1	9	1	8	1	7	1	6	1	5	1	4	1	3	ī	2	1	1	1	0	7

1. ZEPO BITS(15-4) OF Px.

2. LOAD PR BITS (3-0) WITH N.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

STL PR.Y

STOPE RI IN MEMORY AT LOCATION Y

ĺ.	PH	1	PL	1	BA	1				OP	ෆ	DE	39					1				Px				1	В	ITS	(19-	16)	OF	Y	_
į	PH	1	PT,	1												BIT	°S (1	5-0	) 0	FY							_				-		_
ï	17	1	16	-	15	1	14	13	1	12	1	11	1	10 1	9	1	8	1	7	1	6	1	5	1	8	17	3	1	2	7	_	-	_

1. STORE Rx IN MEMORY AT LOCATION Y.

2. Px IS UNCHANGED.

APPPOXIMATE EXECUTION TIME 3.75 MICROSECONDS

STAL Px, Y, RA STOPE Rx IN MFMOPY AT LOCATION Y AND SET RA TO Y

PH	1	PL	- 1	BA	-1	OP	CODES	3A/3B			Rx	- 1	BITS (19-16)	OF	Y
H	1	PL							BITS (15	OI OF Y					

1. STOPE PX IN MEMORY AT LOCATION Y AND SET RA TO Y.

2. Rx SHOULD NOT BE EQUAL TO EITHER MEMBER OF THE RA REGISTER PAIR.

3. Rx IS UNCHANGED.

NOTE: BIT 6 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR PA=12 AND 1 FOR RA=14. BITS(15-4) OF GENERAL REGISTER 12/16 ARE NOT CHANGED BY THIS INSTRUCTION. SEE NOTE CONCENNING RA ON PAGE B2.

APPROXIMATE EXECUTION TIME 4.05 MICROSECONDS

ST Rx, N (RA)

STOPF Px IN MEMORY AT LOCATION RA INDEXED BY N

í	РН	1	P	L	1	BA	1	 _			_	OP	COE	es	48/	189				_	1				Rx				1				N				7
i	11	7 1	_	16	ī	15	-1	14	1	13	1	12	1	11	1	10	1	9	- 1	8	1	7	1	6	1	5	1	8	ī	3	1	2	1	1	1	0	7

1. STORE Rx AT LOCATION DETERMINED BY ADDING N TO RA.

2. Rx IS UNCHANGED.

NOTE: BIT 8 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR RA=12 AND 1 FOR PA=14. SEE NOTE CONCERNING PA ON PAGE B2.

APPROXIMATE EXECUTION TIME 2.70 MICROSECONDS

STA Rx, N (RA)

STOPE Rx IN NEMOPY AT LOCATION RA INDEXED BY N AND UPDATE RA

į	P	R	1	PL	1	BA	1			CF	co	DES	48	/4B			- 1				Rx				1				10	_	_		٦
i	_	17	1	16	- 1	15	1	18	13	1 1	2	1	1 [	10 [	9	1	3	7	ī	6	1	5	ī	4	1	3	1	2	T	1	1	0	7

STORE PX AT LOCATION DETERMINED BY ADDING N TO RA AND UPDATE RA.
 RX SHOULD NOT BE EQUAL TO EITHEP MEMBER OF THE RA REGISIER PAIR.

3. Rx IS UNCHANGED.

NOTE: BIT 8 IS DETEPHINED BY THE VALUE OF RA. IT IS 0 FOR RA-12 AND 1 FOR RA-14. BITS(15-4) OF GENERAL REGISTRY 12/14 ARE NOT CHANGED BY THIS INSTRUCTION. SEE NOTE CONCEPTING RA ON PAGE B2.

APPPOXIMATE EXECUTION TIME 2.70 MICROSECONDS

STX Rx, Ry (RA)

STOPF RX IN MEMORY AT LOCATION RA INDEXED BY RV

į	PH	1	1	PT.	1	10	A	1		_		_	OP	co	DE	s	BC/	'AD					1		_		Rx	-	_		1		_	_	Ry	_	-	-	7
	1				1		15	1	14	1	13	1	1	2 1		11	1	10	1	9	1	8	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	_

1. STORE Px AT LOCATION DETERMINED BY ADDING RY TO RA.

2. Rx IS UNCHANGED.

NOTE: BIT 8 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR PA=12 AND 1 FOR RA=14. SEE NOTE CONCEPNING PA ON PAGE 82.

APPROXIMATE EXECUTION TIME 2.70 MICROSECONDS

STAX Rx, Ry (RA)

STORE RX IN MEMORY AT LOCATION RA INDEXED BY RY AND UPDATE RA

[ PI	1	PL	-1	BA	1				OP	COD	ES	4E	42					1				Rx			1			Ry			- 1
1 3	7 1	16		15	1	19 1	1	3 1	12	-	11	1.1	20	) 1	0	-	8	-	7	-	6	-	- 6	_	-	7	-	 -	 -	-	٠,

- 1. STORE Px AT LOCATION DETERMINED BY ADDING RY TO RA.
- 2. Rx SHOULD NOT BE EQUAL TO EITHER MEMBER OF THE RA REGISTER PAIR.
- 3. Rx IS UNCHANGED.

NOTE: BIT 8 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR RA=12 AND 1 FOR RA=18. BITS (15-4) OF GFMERAL REGISTER 12/19 ARF NOT CHANGED BY THIS INSTRUCTION. SEF NOTE CONCEPNING RA ON PAGE B2.

APPROXIMATE EXECUTION TIME ' 2.70 MICROSECONDS

STAP N (PA)

STOPE ACCESS FUNCTION USING PEGISTER 0 AT LOCATION RA INDEXED BY N

PH	1	Pī,	-1	BA	1				4	OP	CODE	70								0	/1				1			26		
PH	-	Pf,	0.3	BEC0	1	CW0	BDS	R1	BDS	ROI	ISO 1	11	1500	10	PD1	IUPD	0	IDL1	IID	LO	1	<b>PW1</b>	ī	PWO	110	121	1MM2	0 1	10111	1100
17	-	16	-	15	7	7.81	1 1	13 1	1	2 1	11	-	10	-	9		_	7				-	-		-	2		_		-

- 1. COMPUTE THE EFFECTIVE ADDRESS FOR THIS INSTRUCTION BY ADDING N TO RA.
- 2. SAVE THE CONTENTS OF THE MAIN MEMORY STATUS REGISTER.
- LOAD THE MAIN MEMOPY STATUS REGISTER FROM THE SECOND WORD OF THIS INSTRUCTION.
   THP ISOLATE BITS ARE LEFT AS THEY WERE ON ENTRY.
- 4. PERFORM THE INDICATED READ/WPITE OPERATION WISING REGISTER 0 AS DESTINATION/SOURCE FOR THE DATA.
- 5. PARITY IS CORPECTED ON THE DATA PECEIVED FROM THE STORE BEFORE IT IS PLACED IN REGISTER 0.
- 6. ZERO THE CF.
- 7. IF A STOPE ERPOR C OCCURS:
- A. THE CF IS SET EQUAL TO ONE.
  B. THE ER IS CLEARED.
- 9. PESTORE THE PREVIOUSLY SAVED CONTENTS OF THE MAIN MEMORY STATUS REGISTER.
- THIS INSTPUCTION REEPS AN INTERNAL TIMER TO PREVENT THE CC PROM HANGING.
   IF THE INSTPUCTION TIMES OUT, REGISTER 0 IS SET TO ALL ONES.
  - NOTE: THIS INSTRUCTION SHOULD NOT BE CODED DIPECTLY, IT WILL BE SUPPLIED BY SYSTEM MACROS.
  - REPER TO SACC COMMON SYSTEM PROGRAMMEPS GUIDE X-78292.

    NOTE: BIT & IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR RA-12 AND 1 FOP TA-18.
  - SEE NOTE CONCEPNING RA ON PAGE B2.

APPPOYIMATE EXECUTION TIME 9.45 MICROSECONDS.

	D.I.	-	-		_	-									·			
i.	PH	_	PI.	1 BA	1		OF CODE	7C			1	2	/3		1	Ŧ	R M	
	PH	1	Pī,	(BECO	1	CW0	BDSR1 PDSP0 ISO1	(ISO0	(UPD1	[ UPD0	(IDL1	[ IDLO	RW1	1 RWO	(199121	(MH20	120911	[MM10

- 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
- 1. COMPUTE THE EFFECTIVE ADDRESS FOR THIS INSTRUCTION BY ADDING RX TO RA.
- 2. SAVE THE CONTENTS OF THE MAIN MEMORY STATUS REGISTER.
- 1. LOAD THE MAIN MEMORY STATUS REGISTER FROM THE SECOND WORD OF THIS INSTRUCTION.
- THE ISOLATE BITS APE LEFT AS THEY WERE ON ENTRY.
- 4. PEPFORM THE INDICATED PEAD/WPITE OPERATION USING REGISTER 0 AS DESTINATION/SOURCE FOR THE DATA.
- 5. PARITY IS COPPECTED ON THE DATA RECEIVED FROM THE STORE BEFORE IT IS PLACED IN REGISTER 0.
- 6. ZERO THE CF.
- 7. IF A STOPE ERROR C OCCURS: A. THE CF IS SET EQUAL TO ONE.
- A. THE CF IS SET EQUAL TO ONE. B. THE ER IS CLEARED.
- 8. RESTORF THE PREVIOUSLY SAVED CONTENTS OF THE MAIN MEMORY STATUS REGISTER.
- THIS INSTRUCTION REEPS AN INTERNAL TIMER TO PREVENT THE CC FROM HANGING.
   IF THE INSTRUCTION TIMES OUT, PEGISTER 0 IS SET TO ALL ONES.

NOTE: THIS INSTPUCTION SHOULD NOT BE COMED DIPECTLY, IT WILL BE SUPPLIED BY SYSTEM MACROS. REFEP TO JACC COMMON SYSTEM PROGRAMMERS GIJIDS K-74292.
NOTE: BIT % IS DETFEMINED BY THE VALUE OF RA. IT IS 0 FOR %A-12 AND 1 FOR RA-14.
SEE NOTE CONCEPTING RA ON PAGE 3

APPROXIMATE EXECUTION TIME 9.30 MICROSECONDS.

STM Px, N(PA), M INSERT Px UNDER MASK INTO MEMORY AT LOCATION PA INDEXED BY N

Į. P	я	1	PT.	1	PA		t			_	_	OP	col	DES	0	3/0	)5					1				Roc				1				19				-
! P	Ħ	1	PL	1											Ī							IAS1																Ī
1	17	1	16	- 1	1	5	1	14	1	13	1	1:	1	1	1	1	10	1	9	1	8	1	7	1	6	1	5	1	4	1	3	-	2		1	-	0	

- 1. ADD N TO PA TO DETERMINE LOCATION OF WORD WY.
- 2. LOAD GENERAL REGISTER 0 WITH THE CONTENTS OF WORD WY.
- 3. (WY . -MASK) ( (Rx . MASK) --> WY

NOTE: BIT 8 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR RA=12 AND 1 FOR RA=14. SEE NOTE CONCERNING PA ON PAGE B2.

APPROXIMATE EXECUTION TIME 5.40 MICROSECONDS

STVM -Rx , M (PA) INSERT Rx UNDER VARIABLE MASK INTO MEMORY AT LOCATION RA INDEXED BY M

P	Ħ	1	PL	1	BA	1				_	CP (	COD	ES	66,	67	_		-		1		_		Rx				1			_	14		-		7
-	17	1	10	1	15	1	14	1	13	1	12	1	11	1	10	1	9	-	8	1	7	1	-6	1	5	1	4	1	3	T	2	1	1	1	0	~

- 1. OBTAIN MASK FROM THE CONTENTS OF PEGISTER 0.
- 2. ADD N TO PA TO DETERMINE LOCATION OF WORD WY.
- 3. LOAD GENERAL REGISTER 0 WITH THE CONTENTS OF WORD WY.
- 4. (WY · ~MASK) | (Rx · MASK) ==> WY

NOTE: RIT 8 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR RA=12 AND 1 FOR RA=14.
SEE NOTE CONCEPNING PA ON PAGE 82.

APPPOXIMATE EXECUTION TIME 5.55 MICROSECONDS

### PETISTER HOLD/GET ORFRATIONS

### LAYOUT OF 16 WORD HOLD-GET AREA

24	1	PŢ.	1	OP	1			WOR	DS	0 1	WD 1	ARE	PE	SERVED F	OR	RETUR	N ADI	DRES	ss .		- 1	BIT	s (	[19-	16)	OF	RA
PH	1	PŢ	1							_		E	BIT	S(15-0)	OF	RETUR	N ADI	ORE	ss ·								
PR	1	PT.	1					WOPD	2	RE	EDAE	POF	8	REGISTER	2	WHEN	HELD	BY	HOLD	ALL	INST	PUCT	TOP	F			
RH	1	рŧ,	1					WORD	3	RES	ERVE	POF	,	REGISTER	3	WHEN	HELD	BY	HOLD	ALL	INS	RUCT	TO	4			
RĦ	1	PI.	1					WORD	4	RES	SEPVE	D FOR	2	REGISTER	4	WHEN	HELD	BY	HOLD	ALL	INS	TRUCT	101	g.			
PH	-1	PL	1				,	WORD	5	RES	SERVE	D POF	2	REGISTER	5	WHEN	HELD	BY	HOLD	ALL	INS	TRUCT	TOI	1			
RĦ	1	Pt	1					WORD	6	RE	SERVE	D FOF	2	REGISTER	6	WHEN	HELD	BY	HOLD	ALL	INS	TRUCT	OI	M			
RH	1	PL	1					WORD	7	RE	SER VE	D POS	R	REGISTER	7	WHFN	HELD	BY	HOLD	ALL	INS	TRUCT	rioi	10			
RH	1	PL	1					WORD	8	RE	SEDVE	D POF	2	REGISTER	8	WHEN	HELD	BY	HOLD	ALL	INS	TRUCT	TIO	14			
PH	1	PL	ι					WORD	9	RE	SEPVE	D POI	R	REGISTER	9	WHEN	HELD	BY	HOLD	ALL	INS	TRUCT	TIO	10			
RH	1	Pt,	1					WORD	10	RE	SERVE	D POI	R F	EGISTER	10	WHEN	HELD	BY	HOLD	ALL	INS	TRUC	TIO	14			
PĦ	-	PT,	1					WORD	11	RE	SERVE	D FO	RF	REGISTER	11	WHEN	HELD	BY	HOLD	ALL	INS	TRUC	rio	54			
RH	1	Pt	1		_	_		WORD	12	RE	SER VE	D FOI	R F	REGISTER	12	WHEN	HELD	BY	HOLD	ALL	INS	TPUC	rio	N			
PĦ	-	Pī	1		-		_	WOPD	13	RE	SERVE	D POI	R 5	EGISTER	13	WHEN	HELD	ВУ	HOLD	ALL	INS	TRUC	110	10			
RĦ	-	PL	1					WORD	14	RE	SERVE	D FO	9 5	REGISTER	14	WHEN	HELD	BY	HOLD	ALL	INS	TRUC	110	14			
RĦ	1	PL	1					WORD	15	RE	SEPVE	D FOI	RF	REGISTER	15	WHEN	HELD	BY	HOLD	ALL	INS	TRUCT	TO	14			
12	7 1	16	1	15	1	14	1	13 1	1	2 1	11	1 1	0 1	9 1	8	1 7	1	6	1 5	1	0 1	3	1	2	1	1	1

NOTE: WORDS 0 AND 1 ARE HELD BY BSA INSTRUCTIONS AND INTERRUPTS.

HOLD REGISTERS 2 THROUGH 15 IN WORDS 2 THROUGH 15 OF HOLD-GET AREA

P	Ħ	1	RI	L	1	BA	1		_		OP	COD	E 73		-	_			1	_			1		_		1	-		-	0		_	_	7
_	17	1	10	16	1	15	5 1	14	1	13	12	1	1.1	10	1	9	1	8.	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	ï

- 1. STORE EACH REGISTER IN ITS ASSOCIATED WORD OF THE HOLD-GET ARFA.
- 2. REGISTERS STORED ARE UNCHANGED.

NOTE: WORDS 0 AND 1 OF THE HOLD-GET AREA ARE RESERVED FOR RETURN ADDRESS.

APPROXIMATE EXECUTION TIME 28.55 MICROSECONDS

PH PK.N

HOLD PX IN WORD H OF HOLD-GET AREA

PH	1	Pt,	В	A 1				. 0	PC	75 acc						1				Px				1				М	_			1
1 17	1	16	1	15	14	1	13	12	1	11	10	1	9	1	8	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	1

- 1. STORE PX IN WORD N OF HOLD-GET APFA.
- 2. Px IS UNCHANGED.

APPROXIMATE EXECUTIOF TIME 2.70 HICPOSECONDS

GET PEGISTERS 2 THROUGH 15 FROM WORDS 2 THROUGH 15 OF HOLD-GET AREA

PH   PT   BA	OP CODE 73	1	0	1	0 !
1 17 1 16 1 15 1	18 4 13 4 12 4 11 1 10 1	9 1 8 1 7 1	6 1 5 1	4 1 3 1	2 1 1 1 0 1

1. LOAD EACH REGISTER FROM THE ASSOCIATED WORD OF THE HOLD-GET APEA.

NOTE: WORDS 0 AND 1 OF THE HOLD-GET APEA ARE PESERVED FOR RETURN ADDRESS.

APPROXIMATE EXECUTION TIME 26.55 MICROSECONDS

GN Px,N

GET Px FROM WOPD N OF HOLD-GFT AREA

PH	-	P	ī.	1 1	BA	1			OP	C	ODE 7						1				Px				1				N				_1
1 1	7 1		16	1	15	1	18 [	13	12	F	11 [	1	0 1	9	1	8	4	7	1	6	1	5	-1	4	1	3	1	2	-1	1	1	0	1

- 1. LOAD Rx FROM WOPD N OF HOLD-GET APEA.
- 2. WORD N IS UNCHANGED.

APPROXIMATE EXECUTION TIME 2.55 MICROSECONDS

# PEGISTER TO REGISTER OPERATIONS

LR RE, Ry

LOAD RX WITH THE CONTENTS OF RY

| PN | PL | RA | OP CODE OC | Rx | Ry | 17 | 16 | 15 | 4 | 3 | 2 | 1 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 10

1. LOAD RE WITH THE CONTENTS OF RY.

2. Ry IS UNCHANGED.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

LPM Rx,Ry,N LOAD Rx WITH THE CONTENTS OF RY UNDER IMMEDIATE MASK, M

Ĺ	PR		P	t		BA	1						0	P	מס	E 1	D		-		_		- 6				Rx				1				Ry				_
Ĺ	PR	1	P	L	1									_						-	-	N	ASI				_				-	_	-		_	_			-
ī	17	1	_	16	1	15	1	14	1	1	3 [	1	12	1	11	1	1	10	1	9	1	8	1	7	1	6	-	5	1	4	1	3	-	2	-	1	-	0	-

1. Ry . NASK --> RE

2. Ry IS UNCHANGED.

APPROXIMATE EXECUTION TIME 2.40 MICROSECONDS

IRM Rx,Ry,R INSERT RY INTO Rx UNDER INHEDIATE MASK, M

PH   PL   BJ	1	OP CODE 1C	1	Px	F	Ry
PH   PL			NASK			

| 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

1. (Ry · MASK) | (Rx · -MASK) ==> Rx.

2. Ry IS UNCHANGED.

APPROXIMATE EXECUTION TIME 2,40 MICROSECONDS

EXR Rx, Ry EXCHANGE THE CONTENTS OF RX WITH THE CONTENTS OF RY

P	N I	1 8	飞	1	BA	1				OP	CODE	01	:					1		1	Px			1				Ry		-	7
	17 (	1	16	1	15	1	14	13	11	12	1 11	-	10	-	9	7	A	-	7	6	-	5	b		2	-	2	-	-	_	-

1. EXCHANGE THE CONTENTS OF RX WITH THE CONTENTS OF RY.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

ZP Rx SEPO Rx

_	_	_			_					_		_		 																
I PH	-1	PL	1	BA	1			OP	CODE	06				1				Px				1				0				7
1 17	1	16	1	15		14 1	13 1	12	1 11	1	10 4	9	-	-	7	_	4		-	-	-	-	_	-	-	_	-	-	-	-

1. SEPO REGISTER RE. THE CF IS UNCHANGED.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

### SPECIAL PEGISTEP OPERATIONS

THE FOLLOWING TABLE DEFINES THOSE REGISTERS WHICH ARE ACCESSED WITH THE SPECIAL REGISTER INSTRUCTIONS

PS	TO FIELD FUN	CTION .	FROM FIELD FUNCTION
0	GB ==> MCHTP	22 BIT PATH	TI ==> GB 18 BIT PATH PL=0,PE=0
1	GB ==> SAP	22 BIT PATH	SAR ==> GB 22 BIT PATH
2	GB ==> PA	22 BIT PATH	PA ==> GB 22 BIT PATH
3	GB **> MCHB	22 BIT PATH	MCHB ==> GB 22 BIT PATH
- 8	UNASSIGNED		MMSP ==> GB 20 BIT PATH PL=0,PH=1
5	GB ==> AK	22 BIT PATH	AK ==> GB 22 BIT PATH
6	GB ==> AI	22 BIT PATH	AI ==> GB 22 BIT PATH
7	GB ==> DK	18 BIT PATH	DK ==> GB 18 BIT PATH
8	GB ==> CI	18 BIT PATH	DI ==> GB 18 BIT PATH
9	GB a=> DB	22 SIT PATH	DB ==> GB 22 BIT PATH
10	GB *=> ER	22 BIT PATH	ER ==> GB 22 BIT PATH
11	GB ==> DB IF DISPLAY BI	22 BIT PATH IT IN SS IS 1	UNASSIGNED
12	GB ==> IM	18 BIT PATH	IM ==> GB 18 BIT PATH
13	GB ==> SS_S	A 1 SETS SS	IS *=> GB 18 BIT PATH
14	GB ==> MS	18 BIT PATH	MS **> GB 18 BIT PATH
15	GB ==> SS_P	A 1 RESETS SS	SS> GB 22 BIT PATH PLCC, PH-CC

NOTE: THE SACC COMMON SYSTEM PROGRAMMEPS GUIDE X-74292 SHOULD BE CAREFULLY STUDIED BEFORE USING ANY OF THE SPECIAL INSTRUCTIONS.

LRS Rx, Ps

LOAD RX WITH THE CONTENTS OF RE

PH PL   BA	OP CODE 22	1	Pac	1	Re
1 17 1 16 1 15 1 18 4	12 4 12 4 11 4 10 4 0				

1. LOAD RX WITH THE CONTENTS OF R8
WHERE RX IS ONE OF THE GENERAL PUPPOSE PEGISTERS
AND P IS ONE OF THE 16 BIT SPECIAL REGISTERS
APPROXIMATE EXECUTION TIME 1.20 MICPOSECONDS

LSR Ps.Px

LOAD RE WITH THE CONTENTS OF RE

1	PH	- 1	I	T.	1	BA	1				0	P	CODE	07						1				Re		Т		1	_			Rx			N MONEY	7
-0	17	7		16		19		14	 13	1	12	1	11	1	10	1	9	4	8	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	7

 LOAD PO WITH THE CONTENTS OF PX WHIFE PR IS ONE OF THE GENERAL PUPPOSE REGISTERS AND PR IS ONE OF THE 16 BIT SPECIAL TEGISTERS APPROXIMATE EXECUTION TIME 1.20 MICPOSECONDS ....

						_		_	-																													_
P	'H	1	PL	- 1	BA	-							OP	cor	E	14						1				Ps				1				0				į
1	17	1	16	1	1	5 (	-	14	1	13	1	12	2 1	1	1		10	1	9	1	8	1	7	1	6	1	5	1	4	1	3	1	2	1	- 1	1	0	-1

- 1. GENFPAL PEGISTER 2 BITS(3-0) ==> Re BITS(19-16)
- 2. GENERAL REGISTER 3 BITS(15-0) ==> Ps BITS(15-0)

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

UNPF PS

UNPACKS 20 BIT SPECIAL REGISTER PS TO GENERAL REGISTERS 2 AND 3

PR   PT   BA	OP CODE 15	1	0	1	Rs	- į
1 17 1 16 1 15 1 10 1	12 4 12 4 12 1 10 1	9 1 8 1 7	1 6 1 5 1	h 1 3	1 2 1 1 1 0	

- 1. Ps(19-16) ==> GENEPAL REGISTEP 2 BITS(3-0)
- 2. Rs (15-0) ==> GENERAL PEGISTER 3 BITS (15-0)
- 3. GENERAL PEGISTER 2 BITS(15-4) ARE SET TO REPO BY THIS INSTRUCTION.

NOTE: IT IS NOT NECESSAPY TO SUPPPESS THE GATING BUS PARITY CHECK WHEN USING THIS INSTPUCTION ON A SPECIAL REGISTEP THAT MAY NOT HAVE CORPECT PARITY.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

EPANCH TO LOCATION Y

PH   PT   B	A I	CP CC	DES 56	/57				1						OF	FSF	T						
17 1 16 1	15 1 10	13 1 12	11 1	10	9	1	8	1	7	1	6 1	5	1	4	1	3	1	2	1	1	1	0

1. BRANCH TO LOCATION Y WHICH IS DETERMINED BY ADDING OFFSET TO THE PA.

NOTE: BIT 8 IS DETERMINED BY THE SIGN OF THE OFFSET. IT IS 0 FOP POSITIVE OFFSET AND 1 FOR NEGATIVE OFFSET. APPPOXIMATE EXECUTION TIME 1.80 MICPOSECONDS

BR N (FA)

PRANCH TO LOCATION RA INDEXED BY N

í	PH	1	PI	1	BA	1			-	OP (	200E 55					1			Π	0/1				1				N			_	į
1	1	7 1	1	6 1	15	1	14 (	13	1	12 1	11	10	9	1	8	1	7	1	6	1	5	1	4	1.	3	1	2	1	1	1	0	1

- . BRANCH TO LOCATION Y WHICH IS DETERMINED BY ADDING N TO PA.
- 2. THIS INSTRUCTION IS ONLY CAPABLE OF FORWARD BPANCHES.

NOTE: BIT 4 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR PA=12 AND 1 FOR PA=14. SEE NOTE CONCEPNING PA ON PAGE B2.

APPROXIMATE EXECUTION TIME 2.70 MICROSECONDS

BL Y

PRANCH LONG TO LOCATION Y

PH	1	1	PL	1		BA	1	_	_		 _	_	(	P	0	DE	31	:						-1				0			_	1	1	ITS	(19	-10	5) (	P	!
PH	-	1	PL	1	1		_									_		_				BIT	rs (	15-1	0)	OF	Y					_							
17	7	ī	10	5 1	1	15	1		14	ī	13	ı	12	1		11	ī	1	n	ı	9	1	8	1	7	1	6	ı	5	1	4	1	3	1	2	1	1	1	0

1. BRANCH TO LOCATION Y.

APPPOXIMATE EXECUTION TIME 2.70 MICROSECONDS

BRANCH ON CONDITION TO LOCATION Y

				-
PH PL   BA	·CP CODES 58/59	1	OFFSET	1
16 1 16 1 1	10 . 12 . 12 . 11 . 10 .	9 1 8 1 7 1	6 1 5 1 4 1 3 1	2 1 1 1 0 1

1. IF THE CF IS EQUAL TO:

A. TERO, DO NOT BRANCH AND EXECUTE THE NEXT SEQUENTIAL INSTRUCTION.
B. ONE, BPANCH TO LOCATION Y WHICH IS DETERMINED BY ADDING OFFSET TO THE PA.

NOTE: BIT 8 IS DETERMINED BY THE SIGN OF THE OFFSET. IT IF 0 FOR POSITIVE OFFSET AND 1 FOR NEGATIVE OFFSET. APPROXIMATE EXECUTION TIME 1.95 MICROSPCONDS FOR CF=1 AND 1.50 MICROSECONDS FOR CF=0

ISSUE 1

ī	PH	1	PL	1	BA	1	_				CP	COL	ES	5A	/58				_	1							OF	FSE	т					_		7
-	17	-	16	- 1	15	-	14	1	13	1	12	1	11	1	10	1	9	1	8	1	7	1	6	_	5	1	8	1	3	-	2	1	1	1	0	7

1. IF THE CF IS EQUAL TO:

A. 7ERO, SPANCH TO LOCATION Y WHICH IS DETERMINED BY ADDING OFFSET TO THE PA.

B. ONE, DO NOT BRANCH AND EXECUTE THE NEXT SEQUENTIAL INSTRUCTION.

NOTE: BIT 8 IS DETERMINED BY THE SIGN OF THE OFFSET. IT IS 0 FOR POSITIVE OFFSET AND 1 FOR NEGATIVE OFFSET. APPROXIMATE EXECUTIOF TIME 1.50 MICROSECONDS FOR CF=1 AND 1.95 MICROSECONDS FOR CF=0

BCL Y BRANCH LONG CH CONDITION TO LOCATION Y

PH	1	PL	1	BA	1			_		01	PC	ODE	50	-	_				1				0				1	В	ITS	(19	-16	) 0	FY	_
PH	-	Pf.	1														BI	rs (1	5-0	n c	FY												_	
17	1	16	1	15	5 1	14	1	13	1	12	ī	11	1	10	1	9	1	8	1	7	1	6	1	5	1	4	-	3	1	2	1	1	1	0

- 1. IF THE CF IS EQUAL TO:
  - A. 72PO, TO NOT BRANCH AND EXECUTE NEXT SEQUENTIAL INSTRUCTION. B. ONF, SPANCH TO LOCATION Y.

APPROXIMATE EXECUTION TIME 2.70 MICROSECONDS FOR CF=1 AND 1.80 MICROSECONDS FOR CF=0

BNCL Y BPANCH LONG ON NOT CONDITION TO LOCATION Y

í	PH	1	PL	1	BA	1					01	P (	DDE	51	1					1				0				1	В	ITS	(19	-16)	01	f Y		1
ï	PH	1	Pt,	1														BIT	5 (1	5-0	0	PY														1
1	17	1	16	1	15	1	14	1	13	1	12	1	11	1	10	-	9	1	8	-	7	1	6	1	5	1	16	1	3	1	2	1	1	1	0	1

- IF THE CF IS EQUAL TO:

   A. ZEPO, FRANCH TO EXCATION Y.
   B. ONE, DO NOT BRANCH AND EXECUTE THE NEXT SEQUENTIAL INSTRUCTION.

APPROXIMATE EXECUTION TIME 1.80 MICROSECONDS FOR CP=1 AND 2.70 MICROSECONDS FOP CF=0

BRANCH ON INDEX NOT ZERO TO LOCATION Y BX RX.Y

PH	PL	B/	. 1			OP	CODE	3C						1				Px				1	P	ITS	(19	-16)	OF	٧	
PH	1 Pt	1										BIT	S (1	5-0	0	FY				_									
1 17	1 1	1	15 1	14	13	12	1 11	1	10	1	9	1	8	-	7	-	6	1	5	1	4	1	3	1	2	1	1		0

- 1. IF THE CONTENTS OF Rx IS NOT EQUAL TO ZERO, DECREMENT RX BY 1 AND BRANCH TO LOCATION Y.
- 2. IF THE COMTENTS OF RX IS EQUAL TO TERO, EXECUTE THE NEXT SEQUENTIAL INSTRUCTION. APPROXIMATE EXECUTION TIME 2.55 MICTOSECONDS FOR Rx=0 AND 2.70 MICPOSECONDS FOR Px==0

PH	1	RI.	1	БA	1					E 5				1				2/3			1				Rx			
	7 4	- 1	5 1	15		4.0	13 1	-			10		0	-	7	-	-		6		-	2	-	2		1	-	0

- 1. BRANCH TO LOCATION Y, WHICH IS DETERMINED BY ADDING THE CONTENTS OF Rx TO PA.
- 2. THIS INSTRUCTION IS ONLY CAPABLE OF FORWARD INDEXING.

NOTE: BIT & IS DETERMINED BY THE VALUE OF PA. IT IS 0 FOR PA=12 AND 1 FOF RA=14. SEE NOTE CONCERNING RA ON PAGE B2.

APPROXIMATE EXECUTION TIME 2.55 MICROSECONDS

BPAX TX

BRANCH TO LOCATION PA INDEXED BY Rx

1 8	PR	1	RI	L	1	PA		1							0	P	cor	E	5C							1				C				t				Rx				
1	17	1	,	16	ŧ	1	5	ı	11	1	1	13	-	1	12	1	-	1	1	10	1	9	-	1	3	1	7	1	6	- 1	5	1	4	1	3	1	2	1	1	-1	0	
																																 		-			-	200		mar	ome	n

- 1. BRANCH TO LOCATION Y, WHICH IS DETERMINED BY ADDING THE CONTENTS OF RX TO
- 2. THIS INSTRUCTION IS ONLY CARABLE OF FORWARD INDEXING.

APPPOXIMATE EXECUTION TIME 1.65 MICROSECONDS

RSA V

#### BRANCH TO LOCATION Y AND SAVE ADDRESS

PH	1	RL	1	BA		1					-	P	200	E	32						1				1				-1	E	BITS	(19	-16	) 0	FY	
RH	1	PL	-1	-															BIT	5 (1	5-0	) (	F Y	1												
17	1	10	1	1	5	1	18	1	13	1	12	1	1	1	1	10	1	9	1	8	11	7	1	6	1	5	1	4	-1	3	1	2	1	1	1	0

- 1. DECREMENT THE HOLD-GET COUNTER BY 16 AND STORE THE RETURN ADDRESS IN WORDS 0 AND 1 OF THE HOLD-GET AREA. BITS 4-15 OF HOLD-GET AREA WORD 0 ARE ZEROED.
- 2. TRANSFER TO LOCATION Y.
  - NOTE: THIS INSTRUCTION SHOULD NOT BE CODED DIRECTLY, IT WILL BE SUPPLIED BY A SYSTEM MACRO. REFER TO SACC COMMON SYSTEM PROGRAMMERS GUIDE X-74292.

APPROXIMATE EXECUTION TIME 6.15 MICROSECONDS

BSAI X

# BRANCH AND SAVE ADDRESS INDIRECT

																	_					_	_	-	_					_		_	_	_		_	_			_	_			_
i	PE		1	R1	L	ŧ	BA	1	ī						C	P	201	DE	76							1									X									_!
1	1	7	1		15	1	15	1	_	14	1	1	3	1	12	1	_	11	1	10	1	9	)	1	8	1	7	1	6	1	5		1	4	1	- 3		1	2	1	1	1	0	1

- 1. DECREMENT THE HOLD-GET COUNTER BY 16 AND STORE THE RETURN ADDRESS IN WORDS 0 AND 1 OF THE ROLD-GET AREA. BITS 4-15 OF HOLD-GET AREA WORD 0 ARE ZEROED.
- 2. BRANCH TO LOCATION X+2.
  - NOTE: THIS INSTRUCTION SHOULD NOT BE CODED DIRECTLY, IT WILL BE SUPPLIED BY A SYSTEM MACRO.
    - RFFFR TO SACC COMMON SYSTEM PROGRAMMERS GUIDE X-74292.

APPROXIMATE EXECUTION TIME 5.70 MICROSECONDS

	PH	1	PL	1	БA	1	_		_		01	PC	ODE	50				-	_	1				0				1		_	_	0		_		7
1	1	7 1	11	5 1	15	1	14	1	13	1	12	1	11	1	10	1	9	1	8	1	7	1	6	1	5	1	ц	1	3	1	2	1	1	1	0	ī

1. SET THE CF EQUAL TC:

A. ONE, IF PEGISTEP ( IS ZEPO. B. 7EPO, IF PEGISTEP 0 IS NONZEPO.

2. SET THE CP CODE PIL BIT EQUAL TO BIT 15 IN WORD 0 DP THE HOLD-GET APEA.

- 3. BPANCH TO THE PETUPN ADDRESS WHICH IS STORED IN WORDS 0 AND 1 OF THE HOLD-GET AREA.
- 4. ADD 16 TO THE HOLD-GET COUNTEP.

NOTE: THIS INSTRUCTION SHOULD NOT BE CODED DIRECTLY, IT WILL BE SUPPLIED BY A SYSTEM MACPO. REFER TO SACC COMMON SYSTEM PROGRAMMERS GUIDE X-74292.

APPPOXIMATE EXECUTION TIME 4.95 MICROSECONDS

RTCAN N

LOAD RETUPN CODE AND BRANCH TO SAVED ADDRESS

PH   PL   BA		OP C	ODE 50				1			-	3			1				N			$\equiv$
17   16   15	14 ( 13 (	12	11	10 1	9	1	8 1	7	1	6	5	1	4	1	3	1	2	1	1	1 0	1

1. LOAD THE PETUPN CODE N INTO BITS (3-0) OF PEGISTER 0. BITS (15-4) OF PEGISTEP 0 APE SET TO ZERO BY THIS INSTRUCTION.

- SET THE CF EQUAL TO:

   A. ONE, IF PEGISTEP 0 IS ZERO.
   B. 7FPO, IF PEGISTEP 0 IS NONZEPO.
- 3. SET THE OP CODE FIL BIT EQUAL TO BIT 15 IN WOPD 0 OF THE HOLD-GET APEA.
- 4. BPANCH TO THE PETUPN ADDRESS WHICH IS STOPED IN WORDS 0 AND 1 OF THE HOLD-GET AREA.
- 5. ADD 16 TO THE HOLD-GET COUNTEP.

NOTE: THIS INSTRUCTION SHOULD NOT BE CODED DIRECTLY, IT WILL BE SUPPLIED BY A SYSTEM MACPO. PEPER TO SACC COMMON SYSTEM PROGRAMMEPS GUIDE X-74292.

APPPOXIMATE EXECUTION TIME 5.10 HICPOSECONDS

BTSAG

GET PEGISTEPS 2 THROUGH 15 AND BRANCH TO SAVED ADDPESS

PR   PT.   PA	Of	CODE 5D			1	1	1	1	0		
1 17 1 16 1 15 1 14	13   12	1 11 1	10   9	1 8	† 7	1 6 1	5 1	4 1 3	1 2 1	1 1	0 1

- 1. LOAD REGISTERS 2 THPOUGH 15 FROM WORDS 2 THPOUGH 15 DF THE HOLD-GET APEA.
- SET THE CF EQUAL TC:
   A. ONE, IF PEGISTEP 0 IS ZEPO.
   B. ZEPO, IF PEGISTER 0 IS NONZERO.
- 3. SET THE OP CODE PIL BIT EQUAL TO BIT 15 IN WOPD 0 DF THE HOLD-GET APEA.
- 4. BPANCH TO THE PETURN ADDRESS WHICH IS STORED IN WORDS 0 AND 1 OF THE HOLD-GET AREA.
- 5. ADD 16 TO THE HOLD-GET COUNTEP.

NOTE: THIS INSTRUCTION SHOULD NOT BE CODED DIPECTLY, IT WILL BE SUPPLIED BY A SYSTEM MACRO. REFER TO JACK COMMON SYSTEM PROGRAMMEP® GUIDE X-74292.

APPROXIMATE EXECUTION TIME 31.20 MICPOSECONDS

į	28	1	PL	1.3	BA [			OP C	ODE 50	)				1				2			1				N				j
ī	17	1	16	1	15	14 [	13	12	11.1	10	9	1	8	1	7	-1	6	1	5	1	1	3	1	2	1	1	1	0	1

1. LOAD REGISTERS 2 THROUGH 15 FROM WORDS 2 THROUGH 15 OF THE HOLD-GET APEA.

2. LOAD THE RETURN CODE N INTO BITS (3-0) OF REGISTER 0.

BITS (15-4) OF PEGISTER 0 ARE SET TO ZERO BY THIS INSTRUCTION.

3. SET THE CF EQUAL TO: A. ONE, IF REGISTER 0 IS MERO. B. 7ERO, IF REGISTER 0 IS MONZERO.

4. SET THE CP CODE FIL BIT EQUAL TO BIT 15 IN WORD 0 OF THE HOLD-GET AREA.

5. BRANCH TO THE RETUPN ADDRESS WHICH IS STORED IN WORDS 0 AND 1 OF THE HOLD-GET AREA.

6. ADD 16 TO THE HOLD-GET COUNTER.

NOTE: THIS INSTRUCTION SHOULD NOT BE CODED DIRECTLY, IT WILL BE SUPPLIED BY A SYSTEM MACRO.

REFER TO SACC COMMON SYSTEM PROGRAMMERS GUIDE X-74292.

APPROXIMATE EXECUTION TIME 31.35 HICROSECONDS

PROGRAM INTERPUPT END

ľ	PH	-	1	PL	1	BJ	1	1	_	-	_	_	OP	cor	DE S	D		-	_		1		_		4		-	Ī	1				0				1
ī	1	7	ī	16	- 1	1	15	1	14	1	13	1	12		11 [	1	0 1	9	1	8	1	7	1	6	1	5	1		1	3	1	2	1	1	1	0	1

- 1. ZERO THE BLOCK INTERPUPT (BIW) BIT IN THE SYSTEM STATUS REGISTER WHICH WILL ENABLE INTERRUPTS.
- 2. RESTORE THE OP CODE FILL BIT WHICH WAS SAVED IN BIT 15 OF WORD 0 OF THE HOLD-GET AREA.
- 3. BRANCH TO THE PETURN ADDRESS WHICH IS STORED IN WORDS 0 AND 1 OF THE HOLD-GET AREA.
- 4. ADD 16 TO THE HOLD-GET COUNTER.

APPROXIMATE EXECUTION TIME 4.20 MICROSECONDS

#### APITHWFTIC OPFFATIONS

AP Px, Py

ADD BY TO PK AND STOPE THE PESULT IN RK

PH   PL   BA	OP CODE 03	I Px	1	Pv	
1 17   16   15	14 1 13 1 12   11   10	9   8   7   6   5	8 1 3	1 2 1 1 1	٠

1. ADD THE CONTENTS OF FV TO THE CONTENTS OF RX AND PLACE THE PESULT IN PX.

SET THE CF EQUAL TC:

 A. ONE, WHEN THIS INSTRUCTION CAUSES A CARRY PEYOND B (15).
 F. FERO, WHEN THERE IS NO CAPPY BEYOND B (15).

3. Py IS UNCHANGED.

APPPOXIMATE EXECUTION TIME 1.20 MICPOSECONDS

AI Rx,I ADD 16 BITS OF IMMEDIATE DATA, I, TO RE

	P	H	1	PL	1	BA	1					0	PC	3 CC	07						1				Pχ				ī				0		-		-
i	F	H	1	PL	1																I							•	_								•
1		17	1	16	-1	15	5 1	14	-	13	1	12	1	11	1	10	-1	9	1	8	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	

1. ADC I TO THE CONTENTS OF RX AND STORE THE PESULTS IN Rx.

2. SET THE CF EQUAL TO:

A. ONE, WHEN THIS INSTRUCTION CAUSES A CAPPY REYOND B(15).
B. 7FPO, WHEN THERE IS NO CAPRY BEYOND B(15).

APPROXIMATE EXECUTION TIME 2.40 MICROSECONDS

AN RX.N ADD & BITS OF IMMEDIATE DATA, N. TO RX

PH   PL   PA	OP CODE 19	l Px	ı	M	
1 17 1 16 1 15 1 14 1	13   12   11   10   9	18171615	1 5 1 3 1	2 1 1 1	0

1. ADD P TO THE CONTENTS OF Px AMD STORE THE RESULTS IN Rx.

2. SET THE CF EQUAL TC:

A. ONE, WHEN THIS INSTRUCTION CAUSES A CAPRY BEYOND B(15).
B. TERO, WHEN THERE IS NO CAPRY BEYOND B(15).

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

A1S Y AED 1 TO THE CONTENTS OF MEMORY AT LOCATION Y

PH	1	PT.	1 °B	1		OP	COPE	3E				1			2		ı		BIT	(19	-16)	OF	٧	
Ha	1	PL	1							BIT	S (1	5-0)	OF	Y										
	_				 				 				-	_		 		_						-

1 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 |

1. ADD 1 TO THE CONTENTS OF MEMORY AT LOCATION Y.

?. SFT THE CF EQUAL TC:

A. ONE, WHEN THIS INSTRUCTION CAUSES A CAPPY PEYOND B(15).
B. 7EPO, WHEN THERE IS TO CAPPY BEYOND B(15).

3. PLACE THE SUM IN MEMORY AT ICCATION Y.

APPROVIMATE EXECUTION TIME 5,40 MICROSECONDS

Γ	PH	ī	Pî.	- 1	PA	- 1	1					4	OP	copi	2 0	18					1				Rx				1	28	co	MPL	EME	NT	OF	19	Ę.
ī	17	-	16	1	1	5 [	1	14	1	13	1	13	2	11	1	10	ı	9	1	8	1	7	I.	6	1	5	1	4	ī	3	1	2	1	1	1	0	-
1.	TH	F :	SUBT	PA:	CTI	ON	IS	N	CCO	PI.	ISI	HED	PY	ADE	OIN	TORE	2	8 (	TAMO				N	TO	P×.												

SET THE CF EQUAL TO:
 A. ONE, WHEN Px ≥N IF Nø0.
 B. TERO, WHEN Px <N.</p>

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

SP PK, PV

SUBTRACT BY FROM Px AND STORE THE RESULT IN Px

í	PH	1	PŢ.	1	BA	ı				OP	CODE	OA				-		1				Px				1			9	y		П		7
	4*		16	-	15	-	18	1	13	12	1 11	_	10	1	9	-	8	-	7	1	6	1	5	1	h	1	3	1	2	1	1	-	0	7

1. SUBTRACT THE CONTENTS OF BY FROM THE CONTENTS OF RX AND STORE THE RESULT IN BX.

- 2. SET THE CF EQUAL TO:
  - A. ONE, WHEN PX?RY. B. 7FRO, WHEN RX(RY.
- 3. Ry IS UNCHANGED.

APPROXIMATE EXECUTION TIME 1.35 MICROSECONDS

SI Px,I .

SUBTRACT 16 BITS OF IMMEDIATE DATA, I, FROM Rx

Γ	рН	ī	PL	ı	23		ı					(	P	COD	E	07						1				Rx				ī				0				7
ï	PH	1	PL	-															28	co	MPL	EME	NT	OF	I													
ī	17	1	11	5 1	1	15	ı	14	1	13	-	12	1	1	1		10	T	9	1	8	1	7	-	6	1	5	-1	4	1	3	1	2	T	1	1	0	-

- 1. SUBTRACT I FROM THE CONTENTS OF Px AND STORE THE RESULT IN Px.
  THE SUBTRACTION IS ACCOMPLISHED BY ADDING THE 28 COMPLEMENT OF I TO Px.
- SET THE CF EQUAL TO:
   A. ONE, WHEN Px ≥ I IF I≠0.
   B. ZERO, WHEN Px < I.</li>

APPROXIMATE EXECUTION TIME 2.40 MICROSECONDS

#### LOGIC OPERATIONS

COM Px[,Py] COMPLEMENT RX[RY] AND STORE IN RX

PH	-1	PL	1	BA	1		OP	CODE	18				1			_	Dχ			1		Px	(Ry	1	_	7
						 12 1			_	 _	_	-	 _	_	-			 _	_		 -	-	_	_	-	 

- 1. IF By IS. NOT SPECIFIED, COMPLEMENT RE AND STOPE IN PR.
- 2. IF Py IS SPECIFIED, COMPLEMENT Ry AND STORE IN Px.
- 3. IF Px IS EQUAL TO:
  - A. 7EPO. SET THE CF.
  - B. NCNTEPO, CLEAP THE CF.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

MI RE,I AND 16 PITS OF IMMEDIATE DATA, I, TO Px AND STORE IN Rx

L bit	-	1 1	L	1	PA	1					01	РС	ODE	07						1				Px			1				2				7
PF		1 1	T.	1					_	_									_	I															1
1 1	7	1	16	1	15	-	14	1	13	1	12	1	11	1	10	1	9	1	я	-	7	-	6	1	5	-	-	3	-	2	-	1	4	n	~

- 1. AND 16 PITS OF IMMEDIATE DATA, I, TO Px AND STORE IN Px.
- 2. IF Rx IS EQUAL TO:

  - A. TERO, SET THE CF. B. NONZERO, CLEAR THE CF.

APPROXIMATE EXECUTION TIME 2.40 MICROSECONDS

AND Py TO Px AND STOPE IN Px

PH   PL   BA	OP CODE 18	1	Rx	1	Py	
1 17   16   15   14	13 1 12   11   10	9   8   7	1 6 1 5	1 4 1 3	1 2 1 1 1	0 1

- 1. AND PY TO PE AND STOPE THE PESULT IN PE.
- 2. Ry IS UNCHANGED.
- 3. IF Rx IS FQUAL TO:
- A. ZEPO. SET THE CF.
- B. NONZERO, CLEAR THE CF.

APPPOXIMATE EXECUTION TIME 1.20 MICPOSECONDS

OI Px.I INCLUSIVE OR 16 BITS OF IMMEDIATE DATA, I, TO RE AND STOPE IN RE

PH   PL   BA	OP CODE 07	1	Rx	1	3	
। प्रमा । एउ. ।		1				

- 1 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3
- 1. INCLUSIVE OF 16 BITS OF IMMEDIATE DATA. I. TO RE AND STOPE IN RE.
- 2. IF Px IS FQUAL TO:

  - A. ZEPO, SET THE CF. B. NONZERO, CLEAR THE CF.

APPPOXIMATE EXECUTION TIME 2.40 MICROSECONDS

rab bx . by

INCTUSIVE OF PY TO PX AND STORE IN PX

i	PH	1	-	শ.	1	FA	1					OP	. (	anc	19						1				Px				1				РY				1
ī	1	7 1		16	1	15	-	18	1	13	1	12	1	11	1	10	ı	9	1	8	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	1

1. INCLUSIVE OF PY TO RX AND STOPE THE PESULT IN PX.

2. Py IS UNCHANGED.

. IF Px IS FOUAT TO:

A. 2FPO, SET THE CF. B. MONZEPO, CLEAR THE CF.

APPROXIMATE EXECUTION TIME 1.20 MICPOSECONDS

EXCLUSIVE OR 16 BITS OF IMMEDIATE DATA, I, TO RE AND STORE IN PE VI DK.I

PR   PT   BA	OP CODE 07	1	Px	1	4	
PR   PT.		I				

1 17 | 16 | 15 | 18 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

1. EXCLUSIVE OF 16 BITS OF IMMEDIATE DATA, I, TO RE AND STORE IN RE.

2. IF Px IS FOUAL TO:

A. 7ERC, SET THE CF. B. NCNZEPO, CLEAR THE CF.

APPROXIMATE EXECUTION TIME 2.40 MICROSECONDS

XP PK, Ry EXCLUSIVE OR BY TO PX AND STORE IN RX

RH   RT.	# BA	1	OP	CODE 1A		1	Rx	1	Ry	

1. EXCLUSIVE OF RY TO PX AND STORE THE RESULT IN RX.

2. Py IS UNCHANGED.

3. IF Rx IS FQUAL TO:

A. ZERO, SET THE CF. B. NONZEPO, CLEAR THE CF.

ARPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

CR RE, RY COMPARE By TO FX

																																			_	_	_
	RH	1	PT		1	BA	1					01	P C	ONE	20						1				Px				1				Ry				i
i	17	1	1	6	1	15	1	14	- 1	13	1	12	1	11	1	10	1	9	1	8	1	7	1	6	1	5	-	8	1	3	1	2	1	1	1	0	-

1. COMPARE PR TO RY BIT FOR BIT.

2. SET THE CF TO:

A. ONE, IF ALL OF THE BITS MATCH.
B. 7ERO, IF ONE OR MORE OF THE RIT POSITIONS MISSATCH.

3. Rx AND Ry ARE NOT CHANGED BY THIS INSTRUCTION.

ARPROXIMATE EXECUTION TIME 1.20 MICPOSECONDS

3 2 1

1. PV IS UNCHANGED.

APPROXIMATE EXECUTION TIME 2.70 MICROSECOMES

2. POTATE "x LEFT BY K PIT POSITIONS.

1. THE AMOUNT OF THE FOTATION, N. IS DETERMINED BY THE LOW 4 RITS OF BY.

4. Px IS NOT CHANGED PY THIS INSTRUCTION.

1 17 | 16 | 15 | 14 | 13 | 12 | 11 | 12 | 9 | 8 | 7 | 6 | 4

POTATE PX LFFT AN AMOUNT OFTERMINED BY THE LOW & BITS OF PY OP CODE 12 Px . | PH | PT | BA |

APPPOXIMATE EXECUTION TIME 2.40 MICPOSECONDS

3. SET THE CF TO: A. ONE, IF ALL OF THE BITS COMPARED MATCH.
E. \*PRO. IF ONE OF MORE OF THE COMPARED BIT POSITIONS MISMATCH.

1. COMPAPE PX TO I BIT FOR BIT FOR FVERY BIT IN THE MASK THAT IS ONE. 2. THE BITS IN PX THAT APE COMPAPED APE THE LOW R BITS AFTER THE CONTENTS OF RX HAVE BEEN "OTATED PIGHT RV N.

1 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 |

MASK I

COMPARE 8 BITS OF RX POTATED BY N WITH IMMEDIATE DATA AND MASK 1 PH PT BA I OP CODE 15 PH I PT. I

APPROXIMATE EXECUTION TIME 2.40 MICROSECONDS

A. ONE, IF ALL OF THE BITS MATCH.
B. 7800, IF ONE OR MORE OF THE BIT POSITIONS MISMATCH. 3. Px IS NOT CHANGED BY THIS INSTRUCTION.

1. COMPARE Rx TO I BIT FOR BIT. 2. SET THE CF TC:

PH | PE | BA |

| PH | PT. | BA |

CIRM Rx,I,N,M

PH | PT. | 1 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 |

CI Px.I COMPAPE PX TO 16-BITS OF IMMEDIATE DATA, I

. Px AND Py APE MOT CHANGED BY THIS INSTRUCTION. APPROXIMATE EXECUTION TIME 2.40 MICROSECONDS

2. SFT THE CF TO: A. ONE, IF ALL OF THE BITS COMPARED MATCH.
B. 7EPO, IF ONE OF MOPE OF THE COMPARED BIT POSITIONS MISMATCH.

1. COMPARE PX TO PY BIT FOR BIT FOR EVERY BIT IN THE MASK THAT IS ONE.

OP CODE 15

OP COPE 07

PH I PL I MACK 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 |

.

1

р×

PY

DV

5

ŧ

COMPAPE By TC Px UNDER 16 FIT IMMEDIATE MASK CPW PX.FV.M

FERT PROFIL

# POTATE PR LEFT BY " BIT POSITIONS

-			-	_	-			-	_	-	_	-							-						-	-	***			-			-		-	_		-
1	PH	1	P	P.L	1	PA	-1						C	PC	00	E 1	1					1				ъ×				-1			1	6-N	1			1
1	17	1		16	1	15	1	1	1 1		13	1	12	1	1	1 1	10	1	9	1	8	1	7	1	€	1	5	1	4	1	3	1	2	1	1	1	0	1

1. POTATE PX LEFT PY N PIT POSITIONS.

APPROVIMATE EXECUTION TIME 1.20 MICROSECONDS

pp px, 2y

POTATE Px PIGHT AN AMOUNT DETERMINED BY THE LOW & RITS OF PV

	PH	1	PI	1	E/	1	1			_		C	P	cop	Ε	13						1		П			РX				-				пy				1
ī	1	7 1	1	6 1	-	15	1	14	1	13	1	12	-	1	1	1	10	1	9	- 1	8	1	7		1	6	1	5	1	4	1	3	1	2	1	1	1	0	-

1. THE AMOUNT OF THE POTATION. N. IS DETERMINED BY THE LOW & BITS OF Ry.

2. POTATE PX RIGHT BY N PIT POSITIONS.

3. PV IS UNCHARGED.

APPROXIMATE EXECUTION TIME 1.50 MICROSECONDS

PPH PK.N

#### POTATE PX FIGHT BY N BIT POSITIONS

I	Н	1	P1.	1	BA	1			OP	CODE 11					-1				Px				1				N				į
ī	17	1	16	1	15	1	10 1	13	12 [	11.1	10 [	9	1	8	1	7	1	6	-	5	1	4	1	3	1	2	1	1	1	0	ı

1. POTATE PR RIGHT BY N BIT POSITIONS.

APPROXIMATE EXECUTION TIME 1.20 MICPOSECONDS

T7 9¥

#### TEST PK FCP ALL TEROS

PH	1	PI	1	PA	1				OP (	COD	E 1	8				- 1				Px				-1				Px		
	-	26	-	15	-	10	-	12 1		4		11	0 1	0	8		7	-	6	-	5	1	a.		3	1	2		1	0

1. IF Px IS FQUAL TO:

A. 7FPO, FET THE CF. B. NONZFPO, 7EPO THE CF.

APPROXIMATE FXECUTION TIME 1.20 MICPOSECONDS

FL7 Px.Fv

#### FIND LOW 7FPO IN Px AND RECORD ITS POSITION IN RY

1	PH	-	1	PL	1	2	A	1	-				_	OF	, 0	nne	21		_					9				Rx				1				Py				
ì	1	7	1	16	1	-	15	L	14	1	13	3		12	1	11	1	10	1	9	-	8	3	ı	7	1	6	1	5	-1	8	-	3	1	2	-	1	-1	0	1

1. LOOK THROUGH PX FOR THE FIRST ZERO PROM THE LOW END.

IP A FFPC IS FOUND:
 A. ITS LOCATION IS STANSLATED TO A 4-BIT BINAPY NUMBER WHICH IS PLACED INTO PY, B. PITS(14-8) OF PV APE SET TO TEPO.
 C. "HE CF IS SET TO 1.
 D. THE TEPO IN PR IS SET TO 1.

3. IF A 71RO IS NOT FCURD.
A. THE CF IS SET IC C.
P. TY IS UNCHANGED.

NOTE: PE SHOULD NOT FOURL BY

APPROXIMATE EXECUTION TIME 1.95 MICROSECONDS IF A ZERO IS FOUND AND 1.20 MICROSECONDS IF A ZERO IS NOT FOUND

### SINGLE BIT CPERATIONS

SCF

#### SET THE CONDITION-PLOP

PR   Pt   BA	OP CODE 20	1 0	1	0
1 17 1 16 1 15 1 18 1	13 1 12 1 11 1 10 1 9	1 8 1 7 1 6 1 5	1 0 1 3	1 2 1 1 1 0 1

1. SET THE CONDITION-FLOP.

APPROXIMATE EXECUTION TIME 1.20 MICPOSECONDS

SOP

#### SET OF CODE FIL BIT

1	PI	H	1	PL	1	БА	1	_		1	-		OR	CC	DE	00	,		_	_	_		1				0				ī		_	_	1	_	_		7
	1	17	1	16	1	1	5 1	1	4 1	1	3 1	1	12	ī	11	ī	1	0 1		9	.1	8	1	7	1	6	1	5	ī	45	1	3	1	2	1	1	1	8	1

1. SET OP COTE FIL BIT.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

SBN Px, B

### SFT BIT B IN Px

1	P	н	1	Pt	1	BA	4	_				01	P C	nne	20	:	_	_	_			1				Px				1				В				
1	-	17	ī	16	1	15	1	14	1	13	1	12	1	11	1	1	0	ı	9	1	8	1	7	1	6	1	5	1	4	- 1	3	-	2	1	1	1	0	-

1. SET BIT & IN Px.

2. ALL OTHER BITS OF PX ARE UNCHANGED.

ARPROXIMATEREXECUTION TIME 1.20 MICROSECONDS

ERR RX. R

# SET BIT IN Px DETERMINED BY THE LOW & BITS OF Py

PH   PL   BA	OP CODE 2D	l Rx		Py	
1 17   16   15   14	13   12   11   10   9	1817161	5 1 4 1 3	1 2 1 1 1	0. [

1. THE BIT POSITION B IS DETERMINED BY THE LOW & BITS OF Py.

2. SET BIT B IN Px.

3. ALL OTHER BITS OF RX APF UNCHANGED.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

SPS K(PA) . B

# " SET BIT B IF MEMORY WORD AT LOCATION DETERMINED BY ADDING N TO PA

1	P	н	1	PL	1	BA	ı			CP COD	ES 62/	63				1				В			_	ı				N				j
1	_	17	ī	16	- 1	15	1	14-1	13	12	11 [	10	9	ī	8	1	7	ī	6	1	5	1	43	1	3	-1	2	1	1	1	0	-

1. THIS INSTRUCTION OPERATES ON WORD MY AT LOCATION DETERMINED BY ADDING N TO PA.

2. SET PIT B IN WORD WY.

3. ALL OTHER BITS OF WY ARE UNCHANGED.

NOTE: BIT 8 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR PA=12 AND 1 FOP PA=18. SEE NOTE CONCEPNING PA OF RAGE B2.

ARPROXIMATE FYECUTION TIME 4.35 MICROSECONDS

PH   PT.	BA 1	CP CODE 09	ı	9	1	0	7
1 17 1 16	15   14	13   12   11   10	9 1 8 1 7 1	6 1 5 1	9 1 3	1 2 1 1 1 0	-

1. ZFPO THE CONDITION-FIOP.

APPROXIMATE EXECUTION TIME 1,20 MICROSECONDS

FRN PX.B

ZERC BIT B IN PX

Ε	PH	1	PĹ	1	В	A	1				0	PC	DDE :	24			_	-	1				Rx				1		_		В		_	-	7
1	17	1	10	5 1		15	ı	18 1	13	1	12	1	11	1	10	9	1	8	1	7	1	6	-	5	1	4	1	3	1	2	1	1	1	0	-1

1. ZFRO BIT B IN Px.

2. ALI OTHER BITS OF PR APF UNCHANGED.

APPROVIMATE EXECUTION TIME 1.20 MICROSECONDS

78P Rx. Py

MERO BIT IN PX DETERMINED BY THE LOW 4 BITS OF RY

L	PH	1	PL		1 2	A	1						0	P (	OD	2	25						-				Rx				-1				Py			
1	1	7 1	- 1	6	1	15	T	14	1	1	3	1	12	1	1	1 1		10	ı	9	1	8	1	7	1	6	ŀ	5	1	4	1	3	- 1	2	ī	1	1	0

1. THE BIT POSITION B IS DETERMINED BY THE LOW 4 BITS OF Ry.

2. ZERO BIT P IN Px.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

7BS N (PA) , P ZERC BIT B IN MFMORY WORD AT LOCATION DETERMINED BY ADDING N TO RA

_	_	_		_																											•						_
1 1	PH	1	PI		1	BA	1					CP	COL	ES	64.	/65					-1				В				ı				18				i
1	17	1		6	1	15	1	14	1	1	3 [	1	2 1	1	1	10	1	9	1	8	-	7	T	6	T	5	1	8	1	3	1	2	1	1	1	0	1

1. THIS INSTRUCTION OPERATES ON WORD WY AT LOCATION DETERMINED BY ADDING N TO RA.

2. SERO BIT B IN WORD WY.

3. ALL OTHER BITS OF WY ARE UNCHANGED.

NQTE: BIT 8 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR PA=12 AND 1 FOP PA=14.

SEE NOTE CONCEPNING RA ON PAGE B2.

APPPOXIMATE EXECUTION TIME 4.35 MICROSECONDS

ZERO OP CODE PIL BIT

PH (	Pf.	1	BA	1		OP	CODE	OD				1		0			1		0			
------	-----	---	----	---	--	----	------	----	--	--	--	---	--	---	--	--	---	--	---	--	--	--

1. 7ERO OP CODE FIL BIT.

70P

APPROXIMATE EXECUTION TIME 1,20 MICROSECONDS

ICF PR.N ITSFPT CF IN BIT N OF PK

į	РН	1	PL	1	PA	t					OP	co	DE	30		_	_			1				Rx		_	_	1		_		ы	_	 	7
1	17	1	16	5 1	15	1	14	1	13	1	12		11	1	10	1	9	1	6	-	7	1	6	7	5	,	8	,	3	-	2	,	1	0	۳,

- 1. IF THE CF IS EQUAL TO:
  - A. 7EPO, 7EPO BIT N OF Px. B. ONE, SET BIT N OF Px.
- 2. ALI OTHEP BITS OF Px APF UNCHANGED.

APPROXIMATE EXECUTION TIME 1.20 MICPOSECONDS

### TPPL Rx TEST GENERAL PEGISTEP PARITY LOW

[	PH	1	PT.	1	BA	1				OP (	ODE	5E						1				0		-		1				Px	_	_		٦
- 1	17	1	16	1	15	1	14 (	13 (	1	2	11	1	10	1	9	1	8	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	7

1. SET THE CF EQUAL TO PL OF Rx. PL IS THE PAPITY BIT FOR DATA BITS (7-0) .

APPROXIMATE EXECUTION TIME 1.20 MICPOSECONDS

### TPPH °x TEST GFNEPAL REGISTEP PAPITY HIGH

í	P	H	1	PL		1	BA	1			П			0	P	DDE	51	E	_		_		_	1		_		1				1				Rx	_			7
ı		17	ı	1	6	r	15	1	10	1	_	13	1	12	1	11	1	1	0 1	-	,	1	8	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	٦,

1. SET THE CF EQUAL TO PH OF Rx. PH IS THE PARITY BIT FOR DATA BITS (1578) .

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

# SRPL PR TEST SPECIAL REGISTER PARITY LOW

			_		_	_								_				 												 _
-	PH	4 1	PT.	1	BA	1			OP	CODE	5F					ŧ			0				ŧ				Rs			i
-	17	1	16	,	15		10.1	13 1	12	. 11	,	10 1	-		0	-	7	-		-	-	-	-	-	-	-	-	_	_	 -

1. SET THE CF EQUAL TO PL OF Rs. PL IS THE PARITY BIT FOR DATA BITS (7-0).

NOTE: IT IS NOT NECESSARY TO SUPPRESS THE GATING BUS PARITY CHECK WHEN USING THIS INSTRUCTION ON A SPECIAL PEGISTER THAT MAY NOT HAVE CORRECT PARITY.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

#### TSRPH Rs TEST SPECIAL REGISTER PARITY HIGH

. 1	PE	1	P	- 1	BA	1					Ol	PC	ODE	. SF					_	ı				1				1				Ps	Т			1
-	1	7		16 1	15	5 1	14	t	13	1	12	1	11	1	1	0 1	9	1	8	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	-4

1. SET THE CF EQUAL TO PH OF Rs. PH IS THE PAPITY BIT FOR DATA BITS (19-8).

NOTF: IT IS NOT NECESSARY TO SUPPPESS THE GATING BUS PARITY CHECK WHEN USING THIS INSTRUCTION ON A SPECIAL PEGISTEP THAT MAY NOT HAVE CORRECT PARITY.

APPROXIMATE EXECUTION TIME 1.20 MICPOSECONDS

TEN PX. P

TEST BIT B IF PX

1	PI	1	PL.	ī	ВА	1				OP	CODE 2	8	_			_	1		_	_	Rx				1		_		В			7
1	1	7 [	16	1	15	1	19 [	13	1	12	11 1	10	ı	9	1	8	ı	7	1	6	1	5	1	9	1	3	1	2	1	1 1	0	1

1. SET THE CF FOUAL TO BIT B OF Px.

APPPOYIMATE EXECUTION TIME 1.20 MICROSECONDS

TRP Px, Pv

TEST BIT IN Px DETERMINED BY LOW & BITS OF Py

PH [	PL	1	BA.	1		_		0	PC	DDE	29		_	_	 	1				Rx				1				Ry				
17	16	1	15	1	18	1	13 [	12	ı	11		10				1	7	1	6	ī	5	1	4	ı	3	1	2	1	1	1	0	

1. THE PIT POSITION B IS DETERMINED BY THE LOW & BITS OF Ry.

2. SET THE CE EQUAL TO BIT B OF PK.

APPPOXIMATE EXECUTION TIME 1.20 MICROSECONDS

TEST BIT B IN MEMORY WORD AT LOCATION DETERMINED BY ADDING N TO RA TRS M (PA) , B

1	PI	1	1	PT.	1 2	BA	1				CP (	CODE	ES S	52/	53					1				В			1				N				_!
		17	1	16	1	15	ı	18	13	1	12	1	11	1	10	ı	9	1	8	1	7	1	6	1	5	1	1	3	1	2	1	1	1	0	1

1. THIS INSTRUCTION OPERATES ON WOPD MY AT LOCATION DETERMINED BY ADDING N TO RA.

2. SET THE CF FQUAL TO BIT B IN WORD WY.

NOTE: BIT 8 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR RA=12 AND 1 FOR RA=14. SEE NOTE CONCEPNING RA ON PAGE B2.

APPROXIMATE EXECUTION TIME 3.30 MICROSECONDS

TCC1

TEST CENTRAL CONTROL 1

í	PH	_	1	PL	ī	BA	1					OP	CODI	5 51	,	_	-			1	_	_	_	1				1				P				ì
ì	1	7	ī	16	1	15	1	19	1	13	1	12	11	1	10	ī	9	ı	8	1	7	1	6	1	5	ī	4	1	3	1	2	-	1	.1	0	1

<sup>1.</sup> SET THE CF EQUAL TO:

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

A. 7EPO, IF THIS INSTRUCTION IS EXECUTED IN CC 0. B. ONE, IF THIS INSTRUCTION IS FXECUTED IN CC 1.

SIO

SENC I/O MESSAGE OVER CHANNEL AND SUBCHANNEL DEFINED IN R9

į	peq	1	PL	1	BA	ı			OP (	ODE 2	7				1			_	0			9		_		0		-		ì
ī	1	7 1	1	6 1	15	1	19 1	13 1	12 1	11 1	10 1	_	9 1	8	1	7	1	6		5 1	- 0	-	3		2	-	1		n	7

- 1. IDLE THE MAIN I/O CHANNEL DEFINED IN BITS (15-10) OF REGISTER 9.
- 2. LOAD THE I/O STATUS FEGISTER WITH THE SUBCHANNEL SELECT FIELD DEFINED IN BITS (9-4) OF REGISTER 9 AND
- JAM THE TPANSMIT NCPMAL CONTROL STATE.
- 3. LOAD THE I/O DATA REGISTER FROM REGISTER 10.
- 4. LOAD REGISTER 11 FROM THE I/O DATA REGISTER AND PERFORM A MATCH TEST.
- 5. INITIATE MESSAGE TRANSMISSION.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

SMTO

SENC MAINTENANCE I/O MESSAGE OVER CHANNEL AND SUBCHANNEL DEFINED IN R9

-[	PH	1	PI	- 1	BA	1				0	PC	ODE	27						1			_	2			-	1				0				7
ī	17	7 1	1	6	15	1	14	ī	13	12	1	11	1	10	1	9	1	8	1	7	1	6	1	5	1	4	1	3	1	2	ï	1	1	0	1

- 1. IDLE THE MAIN I/O CHANNEL DEFINED IN BITS (15-10) OF REGISTER 9.
- 2. LOAD THE I/O STATUS REGISTER WITH THE SUBCHANNEL SELECT FIELD DEFINED IN BITS (9-4) OF REGISTER 9 AND JAM THE TPANSMIT MAINTENANCE CONTROL STATE.
- 3. LOAD THE I/O DATA REGISTER FROM REGISTER 10.
- 4. LOAD REGISTER 11 FROM THE I/C DATA REGISTER AND PERFORM A MATCH TEST.
- 5. INITIATE MESSAGE TRANSMISSION.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

TEST FOR I/O MESSAGE IN CHANNEL DEFINED IN R9

í	PH	1	PI		ı	BA	ı			_		OP	COD	E 2	7	_		-		1				9		-		ı			-	0	-		-	7
ī	17	1	1	6	ī	15	1	14	1	13	1	12	1	1 [	10	1	9	1	8	1	7	T	6	ī	5	1	4	1	3	T	2	1	1	1	0	1

- 1. IF A MESSAGE IS PRESENT IN THE MAIN I/O CHANNEL DEFINED IN BITS (15-10) OF REGISTER 9:
  - A. THE CF IS SET TO 1. 8. PEGISTER 11 IS LOADED WITH THE MESSAGE AND CHECKED FOR PARITY. C. THE MAIN CHANNEL IS PUT IN THE IDLE STATE.
- If A MESSAGE IS NOT PRESENT IN THE MAIN I/O CHANNEL DEFINED IN BITS(15-10) OF REGISTER 9: A. THE CF IS SET TO 0.
   B. REGISTER 91 IS UNCHANGED.
- 3. REGISTER 10 IS NOT CHANGED BY THIS INSTRUCTION.

APPROXIMATE EXECUTION TIME 1, 20 MICROSECONDS

											 -							_				_	_	_	_	_	_	_	_	_					_	_
í	PF		I	·L	1	PA	1				0	R C	ODE	27						1				5				1				0				_1
	-	2		16	-	15	-	10	1	13	12	,	11	1	10	1	9	1	a	1	7	1	6	1	5	1	4	1	3	- 1	2	1	1	1	0	1

- 1. IF A MAINTENANCE MESSAGE IS RPESENT IN THE MAIN I/O CHANNEL DEPINED IN BITS (15-10) OF REGISTER 9:
  - A. THE CF IS SET TO 1.

    R. PEGISTFF II IS LOADED WITH THE MESSAGE AND CHECKED FOR RAPITY.

    C. THE MAIN CHANNEL IS PUT IN THE IDLE STATE.
- IF A MAINTENANCE MESSAGE IS NOT RESENT IN THE MAIN I/O CHANNEL DEFINED IN BITS(15-10) OF REGISTER 9: A. THE OF IS SET TO 0.
   PERISTEP 11 IS UNKNAMMED.
- 3. PEGISTER 10 IS NOT CHANGED BY THIS INSTRUCTION.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONDS

TEST THE MAIN I/O CHANNEL DEFINED IN R9 FOR THE IDLE STATE

1	PH	1	Rt.	1	BA	1					OP	COD	8 2	7					1				6				1				0		_		_!
-	17	1	1	1	15	1	14	1	13	1	12	1 1	1 1	10	1	9	1	8	1	7	1	6	1	5	1	8	1	3	1	2	1	1	1	0	1

- 1. IF THE MAIN I/O CHANNEL DEFINED IN BITS(15-10) OF REGISTER 9 IS:
- A. IDLE, FET THE CF TO 1. B. NOT IDLE, SET THE CF TO 0.

APPROXIMATE EXECUTION TIME 1.20 MICROSECONES

710

IDLE THE MAIN I/O CHANNEL DEPINED IN R9

1	PH	1	PL	1	BA	1			OR	CODE	27			_			1				7				1				0				1
į.	17	-	16	1	15	1	18 1	13 1	12	1 11	1	10	1	ġ	1	8	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	1

- 1. IDLE THE MAIN I/O CHANNEL DEFINED IN BITS (15-10) OF REGISTER 9.
  - NOTE: BITS (9-8) OF REGISTER 9 MUST CONTAIN A 3-OUT-OF-6 CODE TO PREVENT AN I/O ERROR.

APPPOXIMATE EXECUTION TIME 1.20 MICROSECONES

### MAINTENANCE AND SPECIAL PUPPOSE OPERATIONS

CONL N(RA) COMPLEMENT WRITE THE ON-LINE STOPE AT LOCATION PA INDEXED BY N

																									_				_			-					_	_
- [	P	Ħ	1	PL	1	BA	1						0	P	ODE	7.8						,1				0/1				1				N				j
i	_	17	1	16	5 1	15	1	11	4	1	13	1	12	1	11	1	11	7 1	9	1	8	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	1

- 1. COMPUTE THE EFFECTIVE ADDRESS FOR THIS INSTRUCTION BY ADDING N TO PA.
- 2. SAVE THE CONTENTS OF THE MAIN MEMORY STATUS REGISTEP.
- 3. READ THE ON-LINE STOPE WITHOUT STOPE FPROP COPPECTION OR PROCESSOP EPROP CORRECTION.
- 4. WPITE THE COMPLEMENT OF THE PECEIVED DATA INTO THE ON-LINE STOPE.
- 5. PESTORE THE PREVIOUSLY SAVED CONTENTS OF THE MAIN MEMORY STATUS REGISTER.
- 6. TERO THE CONDITION FILE FLOP.
- 7. THIS INSTRUCTION KEEPS AN INTERNAL TIMEP TO PREVENT THE CC FROM HANGING. IF THE INSTRUCTION TIMES OUT, PEGISTEP 0 IS SET TO ALL ONES.
  - MOTE: THIS INSTRUCTION SHOULD NOT BE CODED DIRECTLY, IT WILL BE SUPPLIED BY SYSTEM MACPOS.

    PPPER TO 3ACC COMMON SYSTEM PROGRAMMERS GUIDE X-74292.
  - NOTE: PIT 4 IS DETERMINED BY THE VALUE OF DA. IT IS 0 EOR PA=12 AND 1 FOP PA=14. SEE NOTE CONCEPNING PA CN PAGE B2.

APPROVIMATE EXECUTION TIME 9.90 MICROSECONDS.

CONLY Px (PA) COMPLEMENT WRITE THE ON-LINE STORE AT LOCATION RA INDEXED BY Px

																																			_
í	PH	1	Pf-	1	ВА	1				0	PC	SOOE	7A						1				2/3				1				Pχ				j
ì	17	-	16	1	15	1	14	1	13 1	12	1	11	1	10	1	9	1	8	1	7	1	6	1	5	1	4	1	3	1	2	1	1	1	0	1

- 1. COMPUTE THE EFFECTIVE ADDRESS FOR THIS INSTRUCTION BY ADDING Rx TO PA.
- 2. SAVE THE CONTENTS OF THE MAIN MEMORY STATUS REGISTER.
- 3. PEAD THE ON-LINE STORF WITHOUT STOPE ERPOR COPRECTION OF PROCESSOR ERROR COPPECTION.
- 4. WPITE THE COMPLEMENT OF THE PECEIVED DATA INTO THE ON-LINE STORE.
- 5. RESTORE THE PREVIOUSLY SAVED CONTENTS OF THE MAIN MEMORY STATUS REGISTER.
- 6. ZEPO THE CONDITION FIIP FIOP.
- THIS INSTPUCTION KEEPS AN INTERNAL TIMER TO PREVENT THE CC FROM HANGING. IF THE INSTRUCTION TIMES OUT, REGISTER 0 IS SET TO ALL ONES.
  - NOTE: THIS INSTRUCTION SHOULD NOT BE CODED DIPECTLY, IT WILL BE SUPPLIED BY SYSTEM MACPOS.

    REFER TO BACC COMMON SYSTEM PROGRAMMEPS GUIDE X-74292.
  - NOTE: BIT & IS DETERMINED BY THE VALUE OF DA. IT IS 0 FOR RA=12 AND 1 FOR PA=18. SEE NOTE CONCERNING PA ON PAGE B2.

APPROXIMATE EXECUTION TIME 9.75 MICROSECONDS.

PH	1	Pī.	1	BA	ı					nP	cc	DE 7	В					1				0/1			1				34	_			ì
F 17	1	16	1	15	1	14	1	13	1	12	1	11		10	9	- 1	8	-1	7	1	6	1	5	T	1	3	ī	2	1	1	1	0	1

- 1. COMPUTE THE EFFECTIVE ADDRESS POP THIS INSTRUCTION BY ADDING N TO RA-
- 2. SAVE THE CONTENTS OF THE MAIN MPROPY STATUS REGISTER.
- 3. READ THE OFF-LINE STOPE WITHOUT STORE EPPOP CORRECTION OF PROCESSOR EPPOP CORRECTION.
- 4. WPITF THE COMPLEMENT OF THE PECPIVED DATA INTO THE OFF-LINE STOPE.
- S. RESTORE THE REFVIOUSLY SAVED CONTENTS OF THE MAIN MEMORY STATUS REGISTER.
- 6. ZERO THE CONDITION PLIP PLOP.
- 7. THIS INSTRUCTION REEPS AN INTERNAL TIMER TO PREVENT THE CC FROM HANGING.
- IF THE INSTRUCTION TIMES OUT, REGISTEP 0 IS SET TO ALL ONES.
- NOTE: THIS INSTRUCTION SHOULD FOT BE CODED DIPECTLY, IT WILL BE SURFLIED BY SYSTEM MACROS.
  PEFER TO BACK COMMON SYSTEM PROGRAMMERS GUIDE X-74292.
- NOTE: RIT % IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR RA=12 AND 1 FOR RA=1%. SEE NOTE CONCERNING RA ON PAGE B2.

APPPOXIMATE EXECUTION TIME 10.20 MICROSECONDS.

#### COPLX Rx (PA)

COMPLEMENT WRITE THE OFF-LINE STORE AT LOCATION PA INDEXED BY PX

P	PH	1	PL	1	BA	1			OP (	ODE 7	В				1			2/	3			1		-		Px		_		ī
1	17	1	16	1	15	1	18	13	12	11	10 [	9	- 1	8	1	7	1	6 .1	5	-	8	-	3	-1	2	1	1	1	0	ī

- 1. COMPUTE THE EFFFCTIVE ADDRESS FOR THIS INSTRUCTION BY ADDING Px TO RA.
- 2. SAVE THE CONTENTS OF THE MAIN MEMORY STATUS REGISTER.
- 3. READ THE OFF-LINE STORE WITHOUT STORE ERROR COPPECTION OF PROCESSOR ERROR COPPECTION.
- 4. WPITE THE COMPLEMENT OF THE PECEIVED DATA INTO THE OFF-LINE STORE.
- 5. RESTORE THE PREVIOUSLY SAVED CONTENTS OF THE MAIN MEMORY STATUS REGISTER.
- 6. ZERO THE CONDITION FITE PLOR-
- 7. THIS INSTRUCTION LEEPS AN INTERNAL TIMER TO PREVENT THE CC FROM HANGING.
- IF THE INSTRUCTION TIMES OUT, PEGISTEP 0 IS SET TO ALL ONES.
  NOTE: THIS INSTRUCTION SHOULD NOT BE CODED DIRECTLY, IT WILL BE SUPPLIED BY SYSTEM MACROS.
  - ROTF: THIS INSTPUCTION SHOULD NOT BE CODED DIRECTLY, IT WILL BE SUPPLIED BY SYSTEM MACROS.

    PEFFR TO JACC COMMON SYSTEM PROGRAMMERS GUIDE X-74292.
  - NOTE: BIT 4 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR RA=12 AND 1 FOR RA=14. SEF NOTE CONCERNING RA ON PAGE B2.
  - APPPOXIMATE EXECUTION TIME 10.05 MICROSECONDS.

Ε	PH	1	Pī.	1	PA	1					OP	cos	E	01					1			0	/1	1			1			N		_	
į	РН	1	PŢ	11	BECO	1	CM0	PDS	P 1	BDS	RO	ISC	1	{I:	son	UPI	1	1 UP DO	1	InLi	1 II	こし	1	PW1	1	PWO	1996	21	IMM20	18	M11	18	M10
ī	17	1	16	1	15	1	14	1 1	3	1 1	2	1	1	1	10	1 5	_	1 8	-	7	1	6	5	5	7		_	2	1 2	-	1		0

- 1. COMPUTE THE EFFECTIVE ADDRESS FOR THIS INSTRUCTION BY ADDING N TO RA.
- 2. SAVE THE CONTENTS OF THE MAIN MEMORY STATUS PEGISTER.
- 3. LOAD THE MAIN MEMOPY STATUS PEGISTER FROM THE SECOND WORD OF THIS INSTRUCTION.
- THE ISOLATE BITS APE LEFT AS THEY WE'PE ON ENTRY.

  4. PEPPORM THE INDICATED PRAD/MPITE OPERATION HISING REGISTER 0 AS DESTINATION/SOURCE FOR THE DATA.
- 5. PFSTORE THE PREVIOUSLY SAVED CONTENTS OF THE MAIN MEMORY STATUS PEGISTER.
- 6. 7FPO THE CONDITION-FLOP.
- 7. THIS INSTRUCTION KEEPS AN INTERNAL TIMER TO PREVENT THE CC FROM HANGING.
- IF THE INSTRUCTION TIMES OUT, PFGISTFP 0 IS SFT TO ALL ONES.
- 8. CAPTION: THE STOPE OPERATION DEFINED BY THE MAIN MEMORY STATUS CONSTANT IN MOPD 2 MAY ALLOW THIS INSTRUCTION TO GATE STOPE DATA WITH BAD FARITY ONTO THE PROCESSOR GATING BUS. IF HAPDWAPE CHECKS APE FOT INTHIBITED THIS WILL CAUSE A PPOCESSOP SHITCH.
- NOTF: THIS INSTRUCTION SHOULD NOT BE CODED CIPECTLY, IT WILL BE SUPPLIED BY SYSTEM MACPOS.

  PEFFF TO JACC COMMON SYSTEM PROGRAMMER# GUIDE X-74292.
  - NOTE: BIT 4 IS DETERMINED BY THE VALUE OF RA. IT IS 0 FOR PA=12 AND 1 FOR PA=14.

    SEP NOTE CONCERNING PA ON PAGE B2.

APPROXIMATE EXECUTION TIME 7.20 MICPOSECONDS

MSTPX Px (RA) MAIPTENANCE STOPE FUNCTION USING PEGISTEP 0 AT LOCATION PA INDEXED BY 9x

Ε	PH	1	PL	1	BA	1				_	OI	α	nE	01				_		1			2	2/3		_		1			F	×	_	_	
1	PH	1	PT.	1	BECC	1	CWO	BI	DSP	112	DSP	Į IS	101	IIS	00	10	PD1	10	PDO	Į I	DL1	111	DLO	1	PW1	1	PWO	[10	421	IMN	20	110	(11	1 112	110
1	17	1	16	1	15	1	14	1	13	1	12	F	11	1	10	1	9	+	8	1	7	1	6	1	5	ī	4	1	3	1	2	1	1	1	0 1

- 1. COMPUTE THE EFFECTIVE APDRESS FOR THIS INSTRUCTION BY ADDING Px TO PA.
- 2. SAVE THE CONTENTS OF THE MAIN MEMOPY STATUS PEGISTEP.
- 3. LOAF THE MAIN MEMORY STATUS PEGISTER FROM THE SECOND WORD OF THIS INSTRUCTION.
- THE ISOLATE BITS ARE LEFT AS THEY WERE ON ENTRY.
- 4. PEPFORM THE INDICATED PEAD/WRITE OPERATION USING PEGISTER 0 AS DESTINATION/SOURCE FOR THE DATA.
- 5. PESTORE THE PPEVIOUSLY SAVED CONTENTS OF THE MAIN MEMORY STATUS REGISTER.
- . 6. ZERO THE CONDITION-FLOP.
- 7. THIS INSTRUCTION REEPS AN INTERNAL TIMEP TO PREVENT THE CC FROM HANGING.
- IF THE INSTRUCTION TIMES OUT, PFGISTER 0 IS SFT TO ALL ONES.
- 8. CAUTION: THE STOPE OPEPATION DEFINED BY THE MAIN MEMOPY STATUS CONSTANT IN MORD 2 MAY ALLOW THIS INSTRUCTION TO GATE STORE DATA MITH BAD PAULTY ONTO THE PROCESSOP GATING BUS. IF HAPDWAPE CHECKS APE NOT INHIBITED THIS WILL CANCE A PROCESSOP SWITCH.
  - NOTE: THIS INSTRUCTION SHOULD NOT BE COPED DIRECTLY, IT WILL BE SUPPLIED BY SYSTEM MACPOS.

    PEFFP TO JACC COMMON SYSTEM PROGRAMMERS GUIDE x-74292.
  - NOTE: BIT 4 IS DETERMINED BY THE VALUE OF PA. IT IS 0 FOR PA=12 AND 1 FOR PA=14. SEE NOTE CONCEPNING RA ON PAGE B2.

APPROXIMATE FRECUTION TIME 7.05 MICROSFCONDS

1 1	PH   PE.   BA								OP CODE ,17								1							SPAPE													
_	-	H   PT.   X-FIELD										1							Y-FIELD											-							
1	17	1	1	6 [		15 1		14 1		13	1	12	1	11	1	10	1	9	1	8	-1	7	T'	6		5	1	4	1	3	1	2	1	1	1	0	-

- 1. SFT INTEPPPFT MODE.
- 2. GATE X-FIFLE TO THE PIR TO FIELD AND Y-FIELD TO THE MIR FROM FIELD.
- 1. PEPFORM THE INDICATED MICRO OPERATION.
- a. PEPFAT FOR THE NEXT STORAGE MOPP UNTIL MICRO OPERATION WHICH TURNS OFF INTERPRET MODE IS GIVEN.
- S. INTEPPUPTS APP NOT PROCESSED WHEN THE CC IS IN INTERPRET MODE.

MOTE: THIS INSTRUCTION SHOULD NOT BE CODED DIRECTLY, IT WILL BE SUPPLIED BY A SYSTEM MACPO. PEFFR TO JACC COMMON SYSTEM PROGRAMMERS GUIDE X-74292.

APPROXIMATE EXECUTION TIMP 1.20 MICROSECONDS PER MAIN MEMORY ACCESS

MTC

### SINGLE CYCLE MICPO INTERPRET

	PE		
PH   PT. 4 X-FIELD	ELD	 _	

- 1. SET INTERPPET MODE.
- 2. GATE X-FIFLD TO THE MIF TO FIELD AND Y-FIELD TO THE MIF FROM FIELD.
- 3. PERFORM THE INDICATED MICRO OPERATION.
- . CLEAP INTERPRET MODE AND EXECUTE THE NEXT SEQUENTIAL INSTRUCTION.
- 5. INTERPUPTS ARE NOT PROCESSED WHEN THE CC IS IN INTERPRET MODE.

NOTE: THIS INSTPUCTION SHOULD NOT BE CODED DIRECTLY, IT WILL BE SUPPLIED BY A SYSTEM MACRO. PEFFF TO SACC COMMON SYSTEM PROGRAMMERS GUIDE X-74292.

APPROXIMATE EXECUTION TIME 2.40 MICROSECONDS

NOP

#### NO OPERATION

-	_	-	-	_									ODE				_		1		_	_	0				1		_		0	_			7
1	17		16	1	1	5	14		13	1	12	1	11	1	10	9	1	8	1	7	1	6	1	5	1	4	1	3	-	2	1	1	-	0	-
	AP	PRO	XIM	AT	EE	ECU	TIO	N T	IME	1	. 20	MI	CPO	SEC	ONDS														•					٠	

HALT

# HALT THE CENTRAL CONTROL

L	PH	1	PL	-	BA	1			0	PO	ODE 5	D.			1				6	-	-		1	-	_	_	0	_	_		7
1	17		11	5 1	15		14	13 (	12	1	11	10 (	9	8	1	7	1	6	1	5.	1	4	1	3	-		-	1	-	0	

1. THIS INSTRUCTION CAUSES THE CC TO LOOP UNTIL IT IS INITIALIZED OR INTERRUPTED. IF THE CC IS INTERPUPTED THE INTERRUPT WILL PETURN TO THE HALT INSTRUCTION.

• CAUTION: NO CHELK IS MADE TO DETERMINE THE ONLINE/OFFLINE STATUS • OF THE PROCESSOP PRIOR TO EXECUTION OF THIS INSTRUCTION \*

SECRETARY OF THE PROPERTY OF T 200